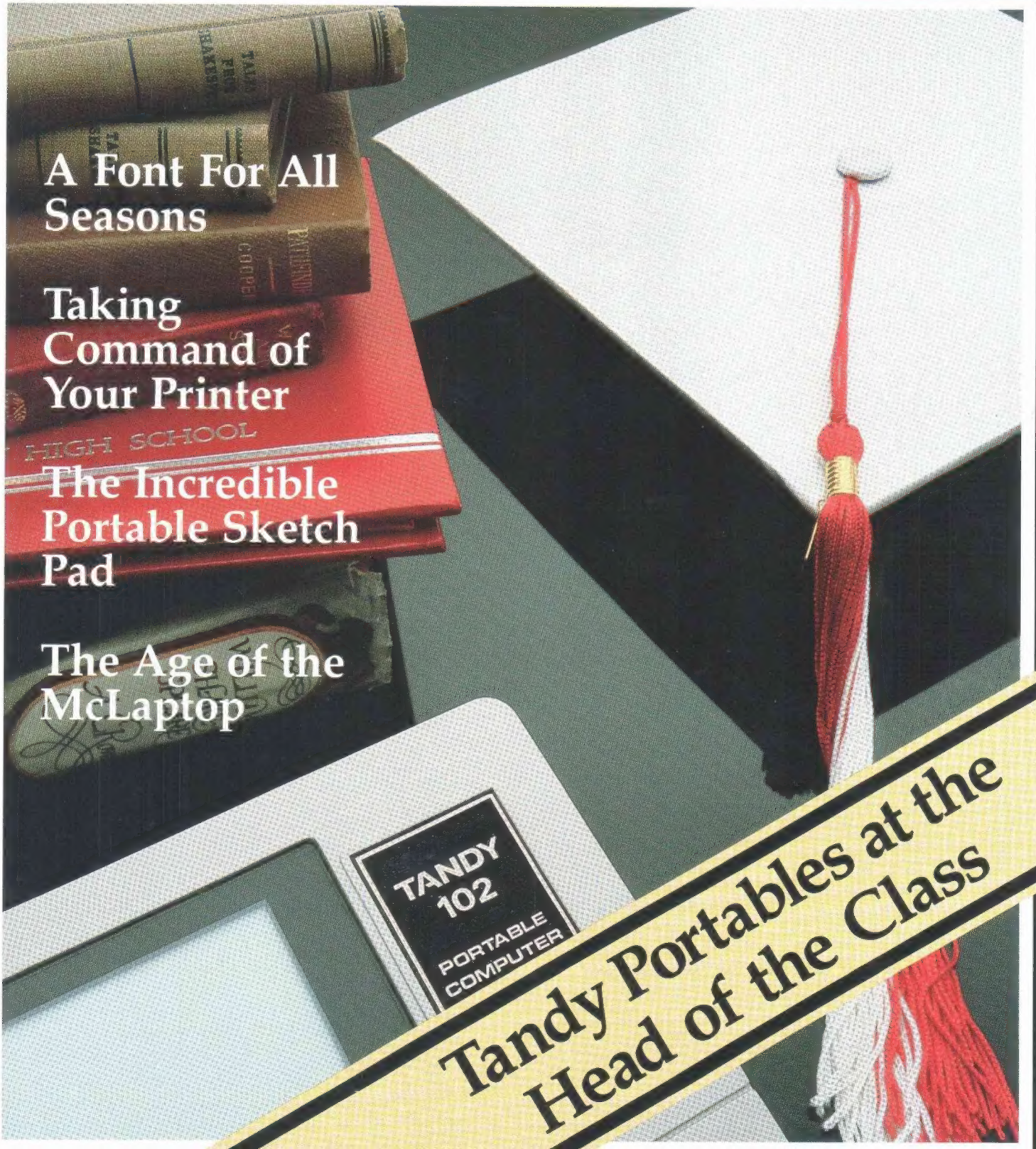
A CW Communications/Peterborough Publication $3.95 /CAN $4.95

portable 100

TANDY LAPTOP COMPUTING VOLUME 4 NUMBER 2 SEPTEMBER 1987



A Font For All Seasons

Taking Command of Your Printer

The Incredible Portable Sketch Pad

The Age of the McLaptop

Tandy Portables at the Head of the Class

Ever Dream of a Faster Modem, Longer Battery Life, and All the RAM Memory and ROM Software you want?

Wake Up! BOOSTER PAK is Here!

Traveling Software introduces the BOOSTER PAK! An incredible breakthrough for Tandy Model 100/102 laptop computers. The BOOSTER PAK gives you an unprecedented 2 megabytes of RAM memory and ROM software. At a very affordable price.

Flexible Design

The BOOSTER PAK is designed to meet your needs now and in the future. You can customize it as you go.

Easy Installation

Just snap the BOOSTER PAK onto the bottom of your Tandy 100/102. Plug in two cables and you're in business.

Built-in Software

Equally as incredible as the hardware is the built-in BOOSTER PAK software. Our unique virtual RAM disk software eliminates any need to do "bank switching." File subdirectories are supported for easy organization of your data. BASIC programs can access files as large as your RAM memory—up to two megabytes! We have also included our popular TS-DOS disk operating system for use with the Tandy disk drive and our new Desk-Link software, as well as an XMODEM program to upload/download files up to 2 megabytes directly to the virtual RAM disk. And to top it off, we have even included a fast action Asteroids game with amazing graphics in our standard BOOSTER PAK.

The BOOSTER PAK Everything You Asked For and More!

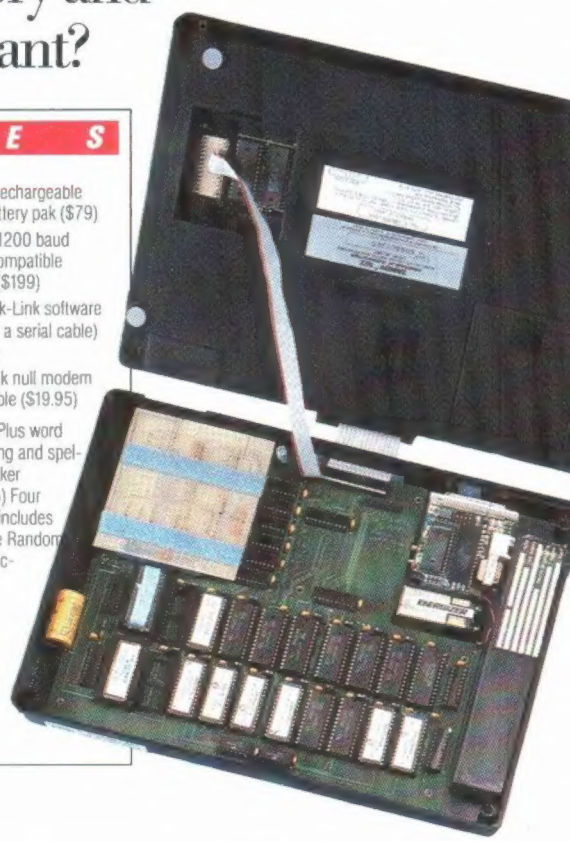
F E A T U R E S

Here's what you get in a standard BOOSTER PAK for only $429:

- 128K of RAM memory (96K available for files)
- 64K of ROM software
- 16 socketed slots for 32K RAM and/or ROM software chips
- 2 Tandy Molex ROM sockets
- Virtual RAM software (no bank switching or copying)
- File folders/sub-directories supported
- TS-DOS disk software built-in
- XMODEM file upload/downloading to virtual RAM
- Long-life lithium battery for RAM backup (rated at 7 years)
- BASIC programming access to virtual RAM
- ROM environment software (mix and match any ROM software)
- Ability to develop ROM software from 32-480K in size
- Fast action Asteroids game (with great graphics) built-in

BOOSTER PAK Options

- 32K RAM (certified low power) chips ($20 ea.)
- 6 slot RAM expansion board ($68)
- 256K RAM expansion modules (requires expansion board) ($159 ea.)
- Internal rechargeable nicad battery pak ($79)
- Internal 1200 baud Hayes-compatible modem ($199)
- IBM Desk-Link software (requires a serial cable) ($39.95)
- Desk-Link null modem serial cable ($19.95)
- Sardine Plus word processing and spelling checker ($199.95) Four chip set includes complete Random House dictionary



Booster Pak™

*To order **BOOSTER PAK** call toll free*

1 800 343-8080

Or write: **Traveling Software**

TRAVELING SOFTWARE

North Creek Corporate Center
19310 North Creek Parkway
Bothell, Washington 98011
206-483-8088

What do you call thousands of words packed into one powerful chip?

The first electronic pocket dictionary for laptop computers.

If you have a laptop computer and you don't have **Sardine**, you're missing out on one of the most powerful laptop tools available today.

It's **Sardine**. The *first* laptop pocket dictionary. The *first* laptop spelling checker. The *first* to be put on a plug-in ROM chip. And the *first* to give you access to 33,000 words from what has been called "the best vestpocket dictionary in the world"—the Random House American dictionary. *(You even get your own printed copy of the dictionary with every* ***Sardine*** *you buy).*

Faster than a desktop!

With **Sardine** and virtually any word processing software, you can look up a single word in seconds or check an entire document up to 25K in size in less than three minutes. All you do is press two keys while you're typing and you can check just the word you're working on or your entire document.

Random House and Wang Labs quality.

Sardine gives you so much more because it gives your laptop computer capabilities you may never have imagined. In addition to the 33,000 word Random House™ American dictionary, over *7,000* of the most frequently used business words are included in the ROM version. These priority words are the results of a two-year, one million word study of business correspondence conducted by Wang Laboratories' lexicographers.

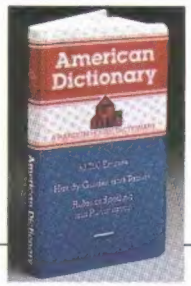
And that's not all. You can also word process with **Sardine**, because it contains the acclaimed T-word word processing program.

Able to create custom dictionaries with a single key.

Just press one key and **Sardine** lets you create an auxiliary, custom dictionary to which you can add words, names, abbreviations or any special terms you use. Add hundreds, even thousands of words, depending on available memory.

F E A T U R E S

Sardine is available for the Tandy Models 100/102, 200, and NEC PC-8201 laptop computers.
Requirements: 16K RAM, Tandy/Purple Computing disk drive optional for ROM version.



- **Sardine** is available on both a 3½ inch disk and a plug-in ROM chip.
- Super fast! Processes over 13,000 dictionary words a minute.
- Look up a single word in seconds. Or, check an entire document up to 25K in less than three minutes. Corrections are automatically inserted.
- ROM version contains the acclaimed T-word word processing program and over 7,000 "priority" words on the ROM chip itself. (ROM version also works with the disk).
- Create your own auxiliary dictionary for special terminology.

P R I C I N G

3½" Disk version $99.95
Plug-in ROM version $169.95
(includes T-Word word processing)

Circle 77 on Reader Service Card

SARDINE

To order ***Sardine*** *call toll free:*

1 800 343-8080

Or write: **Traveling Software**
North Creek Corporate Center
19310 North Creek Parkway
Bothell, Washington 98011
206-483-8088

TRAVELING SOFTWARE

portable 100

VOL. 4, NO. 2 SEPTEMBER 1987

ON THE COVER

FEATURES

DEPARTMENTS

Cover Photograph by Benjamin Magro

Editor
Roger Strukhoff
(603) 924-9471

Managing Editor
Steven F. Smith

Technical Editor
Alan L. Zeichick

Assistant Editor
Janet M. Tiampo

Senior Contributing Editor
Carl Oppedahl

Contributing Editors
Cy Callaghan Emmett J. Carmody
Jay Holovacs Mark Laster
Thomas L. Quindry Ralph Tenny

Design Consultant
Peter D. Koons
High Resolution

Publisher
Jeff DeTray

Associate Publisher
Jim McBrian
(603) 924-9471

Advertising Representative
Carl Cramer
Vertical Marketing Group — (207) 594-6216

Advertising Coordinator
Michelle Boot

Corporate Production Director
Dennis Christensen

Manufacturing Manager
Susan Gross

Assistant Manufacturing Manager
Leslie Walden

President/CEO
Michael S. Perlis

Vice President/Finance
Roger Murphy

Director of Circulation
Frank S. Smith

Circulation Manager
Bonnie Welsh

Single Copy Sales Manager
Linda Ruth

Direct Marketing Manager
Paul Reuss

Telemarketing Manager
Elizabeth R. Kehn

Director of Credit Sales
William M. Boyer

ISSN 0893-942X

Portable 100 is an independent journal not connected with any hardware, software, or peripheral equipment manufacturer. Portable 100 is published by CW Communications/Peterborough, Inc., 80 Elm St., Peterborough, NH 03458. Second Class postage paid at Peterborough, NH and additional mailing offices. Subscription rates in the U.S. are $24.97 for 12 issues. In Canada $45.97 (Canadian). In Mexico, $29.97, U.S. funds drawn on a U.S. bank. Other foreign subscriptions (surface mail) are $44.97 for 12 issues; air mail is $79.97, U.S. funds drawn on a U.S. bank. Address all subscription correspondence to Portable 100, Subscription Dept., 80 Elm St., Peterborough, NH 03458. POSTMASTER: Send address changes to Portable 100, 80 Elm St., Peterborough, NH 03458. *Volume 4, Number 2, September, 1987.*

ROM WITH A VIEW

I Did It Your Way

Go to your local newsstand and scan the magazine racks. Look at the incredible variety of publications available. The most modest store will have magazines about cars, electronics, horses, baseball and other sports, All-Star Wrestling and other entertainment, women, men, young women, young men, babies, families, fashion, and computers. The latter category will include a number of general-interest computer journals as well as a variety of more specific publications. The more tightly focused magazines will be about programming, computer languages, or maybe dedicated to a single computer.

Which brings us to *Portable 100*. Currently not widely distributed—we're working on serving long-suffering subscribers first—*Portable 100* nonetheless has a loyal, okay, fanatical, group of regular readers. Readers take this magazine personally...as well they should, since this magazine belongs to them. Readers not only buy and read this magazine, they help write it.

Check out this issue. There's the usual fine work turned in by Technical Editor Alan Zeichick and Senior Contributing Editor Carl Oppedahl, but there's a lot more to it than that. From a long program on how to discover graphics (page 40) to a printer enhancement (page 33) to a keyboard noise-deadener (page 4) to a Tandy CoCO to Model 100 connector of all things (page 18), *Portable 100* readers have contributed greatly.

This will continue. In future months, we'll have stories about creating menus, indexing with the Super ROM, BASIC text editors, using monitors for video output, and memory conservation...all written by readers.

Articles are edited for grammatical and technical accuracy, of course, and we will work with fledgling authors to get their articles focused correctly and in shape for publication. But remember, this is a readers' magazine. If you have something to contribute, drop us a line. If you see a program that is similar to something you've done, only yours is better, let us know. If you think you're doing something unique with your Tandy laptop computer, don't be quiet about it.

There's a section called "You Be The Editor" on page 49 of this issue, which allows you to rate the articles and the magazine by circling numbers on the reader service card. Take some time to critique us, and while you're at it, circle as many other numbers as you please to get more information about products mentioned in this magazine.

I've been pleased with the response I've received so far to CW Communication's revival of *Portable 100*, and at this writing, the August issue has been in subscribers hands only a few days. I hope this issue is well-received, too.

Remember, it's your magazine. I said it last month, and I'll say it again. Let me know what you think.

Roger Strukhoff, Editor

Low-tech Saves the Day

I would like to avail you of a couple of tricks I've discovered. I believe they improve upon the operating economy and usefulness of two TRS-80 products: the Model 100 and the TRP-100 printer.

I've read that the Model 100's keyboard may be quieted by installing orthodontic rubber bands on the key stalks to cushion the impact of the keys on the keyboard base. While I'm sure this works, I've found another method that I suspect may have certain advantages over using rubber bands.

My method uses the closed-cell foam sheeting that's often used as packing. When cut into one-half inch squares and pierced in the middle, they slip over the key stalk just as a rubber band would. The result is greatly decreased keyboard clatter and a satisfying tactile feedback.

My guess is that the foam would outlast the rubber bands, and probably last the life of the computer, thereby relieving one of the necessity of having to pry the keys off to re-do the bands. One may also be saved the untimely failure of a band.

My other innovation applies to the TRP-100 printer. I picked one up for about $100 when my local Radio Shack Computer Center was clearing them out. I was perfectly happy with the unit until I checked out the Shack's price on thermal paper: $25 for six skinny rolls (minimum purchase). It's no wonder they weren't selling.

I tried a larger roll of thermal paper from a stationary store. But while this roll mounted onto the original roller, the paper failed to feed freely from the roll and invariably jammed. I continued to use the big rolls for awhile, pealing off as much paper as I'd need for printing, then letting the roller draw from the draping, slack length of paper.

Then one day, while perusing through Gemco, I saw what looked as though it might be the key to converting the TRP-100 to use the big rolls. The Brother Roll Paper Holder (model 6600) is designed to snap right onto the Brother EP-44 electronic printer.

This appertenance looked as though it might be adaptable to the TRP-100. When I got home, I was pleased to find out that it matched up almost perfectly with the battery cover of the TRP-100. After drilling a couple of carefully plotted holes, the Brother unit was screwed securely onto the TRP-100's battery cover and voila! The result was a portable printer that accepts widely available, economical, long-play roles of thermal paper...even the color matches.

Brother also markets unbroken rolls of plain paper. But this would have to be used with the ribbon in the printer, and that's another story.

Carl D. Rossman
Upland, Calif.

AN ENTERTAINING PROBLEM

I write TV shows. I was recently writing a MacGyver script for ABC, and here was my problem. My writing partner and I had written a tricky little story about MacGyver defusing a bomb aboard a luxury liner on the high seas. We'd handed in our first draft to the producers at Paramount Studios, and were waiting for our "notes" so we could write the second draft—in Hollywood, like most places, all writing is actually re-writing.

It turned out they wanted to shoot our script on a Monday, at the same time I was to be away on a business trip to New Mexico. I wouldn't be in town to finish making the final script changes on schedule. In fact, I'd only have time to pick up the rewrite notes from the producers before I left. Adding to my dilemma, the producer gave me some hand-typed pages of additional dialogue.

How was I to enter the new material into the computer, do a rewrite on my story in Santa Fe, then deliver the finished script back to Paramount before the week-end was over? I'm sure many people in other sections of the business world have been confronted with similar problems. You have to contact your home office, then deliver or receive information very quickly.

My trusty Model 100 came to the rescue. My little machine is the "old-fashioned," eight line by 40-character LCD vanilla model. I use it all the time. Sometimes it just sits in my office next to my desktop computer and pretends it's a phone dialer, sometimes it's a little calculator, sometimes I use it to make a quick note on a new television or movie idea. For my rewrite crisis, I decided its 32 kilobytes of storage would suffice. If I needed more, I'd just dump it to my portable disk drive.

The first hurdle was the typewritten pages the producer had given me. I needed to get them onto a disk so I could transfer them from my desktop to my Model 100, and I sure didn't want to type them all in by hand. So I stopped by a company in Santa Monica called On Word, which had a great little machine that "reads" typewritten pages and transfers them into any disk format. I quickly had the producer's copy onto a disk for my desktop, downloaded it with Portable Computer Support Group's Disk+ to my 100, then transferred it to 3.5-inch disk on my Tandy drive. I was ready to leave town.

I finished rewriting the script Sunday night. The Monday morning deadline was approaching, and I was still sitting in my Sante Fe motel room. I loaded the new script pages into memory on the Model 100, and called up CompuServe. At the "!" prompt I typed "GO 174" for the Professional area. A menu of options appeared. I picked the correct menu number, type of transfer, etc., and uploaded my revised MacGyver script.

Back in Los Angeles, my partner Tony turned on his computer and downloaded the material from CompuServe onto a disk, then printed it out. Within the hour he delivered it to the producer at Paramount. Mission Impossible—accomplished!

I'll bet you're already thinking of a lot of ways to use this method of transmitting material, if you're not already using it. Naturally, it would have been faster to send my material directly to a computer at Paramount, but

256K RAM FOR THE M102 & M100

Memory doesn't erase when you unplug it —
Buy multiple cartridges and carry them with you anywhere!

Instantly upgrade your Model 102 or your old Model 100 to 256K of RAM or use the budget unit for 160K. That little cartridge simply plugs onto the back of the Model 102. That's all there is to it. Buy several if you like, because they don't lose your files when you unplug them. (For the older type Model 100 we have a special shape that goes inside the little back compartment).

Like eight Model 102's in one.

You then have 8 banks of RAM of 32K each. The additional seven banks also work just like your Main Menu. Push a function key: you are in the second bank. Push again, the third and so forth until you are back to your original bank. Its built-in NiCad battery recharges right from the Model 102 and is guaranteed for a full year.

Copy a file from bank to bank with a function key.

Each bank is like having another Model 102. All the built-in programs and snap-in ROMs appear and work in all banks. Your widebar cursor moves from file to file and you just press ENTER.

Data files in excess of 220K.

You aren't limited to working with banks of 32K each. The ROM software included has extensions to BASIC that allow modification of programs to write and read data throughout the entire 256K. Data files can be created using all available memory, appearing to the user as one continuous 256K of RAM.

More features.

Rename or kill any file. Set the date, day and time with function key ease. Even use non-Radio Shack printers without those tricky dipswitch settings. Whenever you take out the snap-in ROM it leaves a tiny program that lets you switch banks just by pressing ENTER.

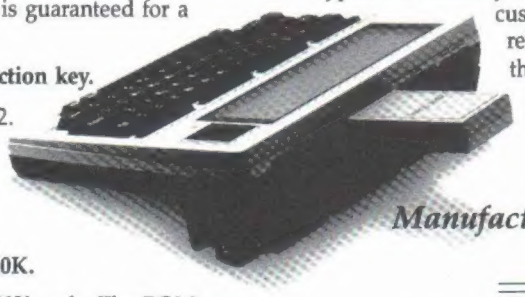
Total RAM*	128K	160K	256K
Model 100	$299	$369	$549
Model 102	—	$369	$549

**Including original 32K*

The complete system.

The small center photo is the new 256K cartridge with the PCSG/Cryptronics 6 ROM bank, that allows six software ROMs installed, plus gives 30 hours of battery power to your computer. Combine that with SuperROM and the Holmes/PCSG portable disk drive (350K on a diskette): the complete portable system.

It is truly an engineering masterpiece. Thousands of PCSG/Cryptronics memory expansions have been shipped to satisfied customers — more are installed and working reliably than any competitive product. We sell these memory expansions on a 30-day trial. Simply return it within 30 days for a full refund if you are not satisfied.



Manufactured by Cryptronics, Inc.

PERSONAL COMPUTER SUPPORT GROUP

11035 Harry Hines Blvd. #207 • Dallas, Texas 75229

214-351-0564

MasterCard VISA

Circle 71 on Reader Service Card.

so far, they're not set up for that type of operation. Other communication services, like The Source, Genie, et al, would have worked as well. I also could have used a BBS.

I now someone out there will ask why no just send the material directly to my partner's machine and skip CompuServe? You're right — next time I'll try it.

Dave Ketchum
Santa Monica, Calif.

Mr. Ketchum's story is an oldie but goodie. His situation is common, we suspect, among readers of Portable 100. *His solution used methods that have been available for some years now, but we thought it interesting to see that even in glittery Hollywood, the mundane Model 100 can be the hero of the story. The Model 100's ability to communicate has always been the key to successful use of its portability. Long after there's a MacGyver, there will be Tandy laptop computers bailing people out of the deadline hoosegow. — Ed.*

ONE FROM THE PAST

I read with interest the (March, 1986) article titled, "SECRET.BA: Protecting .DO Files Through Encryption." It was clearly written except for one very important point. The writer didn't explain how secure the encryption scheme was. One must ask this question because one needs to know what the risks are for storing sensitive data on a computer. I'm not of the belief that the scheme is very secure.

The type of encryption used in the article is based on a scheme called the Vigenere cipher. According to legend, Julius Caesar is credited with using the scheme with a single letter key that offset the alphabet by three letters.

At any rate, there's been a tremendous amount of time for people to come up with ways of breaking this cipher. A motivated person would try to decipher coded messages by looking at the frequency distribution of letters. Additional clues can be gleaned from word size since spaces are not encrypted. If one can guess a word, then one could compute part of the key directly.

I would not encrypt something important using the method described in the article. If you must use the program it would be a good idea to make the following modifications. First, encrypt as many characters as possible. This can be done by modifying the range set in the statements in lines 70 through 91 of the listing. In particular, I would encrypt blanks so word spacing would not betray the key. To do so make the following modifications:

```
70 X = INPUT$(1,1)
 :I = ASC(X)
:IF I<32 OR I>122 THEN 100 ELSE IF
A = "DE" THEN 90
80 I = I + ASC(MID$(K1,J,1))
81 IF I > 122 THEN I = I-91
 :GOTO 81 ELSE GOTO 95
90 I = I-ASC(MID$(K1,J,1))
91 IF <32 THEN I = I+91
 :GOTO 91
```

To conclude, I'd recommend using caution and common sense if you must use the encryption algorithm.

Jeffrey Saltzman
Los Alamos, NM

When we first read Mr. Saltzman's letter we thought he must be in the business of safeguarding nuclear secrets or something. Then we noticed where he is from and realized we were probably right.

—Ed.

ANYONE KNOW?

I'm wondering if your readers can help me. I have a friend who has a Model 100, an Ultimate ROM and a Tandy Portable Disk Drive. She and her husband are missionaries in Papua, New Guinea and will shortly be returning there. They have two young sons whom they are educating at home in New Guinea. I'm trying to locate educational software for the Model 100 that might be able to be used in a remote location to enrich the education of 6- and 9-year-old boys.

If any of your readers have any of this material or could tell me where to obtain it, or if there are any public domain programs that could be used, I would really appreciate learning about it. Thank you for your help in this matter.

Chris Templar
Knoxville, TN

If anyone can help, please contact Chris Templar c/o the Department of Elementary Education, Johnson Bible College, Knoxville, TN 37998, (615) 573-4517. And let us know, too.

—Ed.

WELCOME BACK

It is great that CW Communications/ Peterborough has taken on *Portable 100* magazine! The magazine has been missed dearly by subscribers and advertisers alike...

Richard Echerlin
President
Ultrasoft Innovations
Champlain, N.Y.

I learned from a recent conversation that *Portable 100* magazine is coming back. I was very pleased to hear this and would like to wish you great success.

Approximately one year ago I decided to abandon the software business in favor of running my restaurants. This has left me with copyrights to some valuable software that is just sitting around collecting dust. If you are interested, I could prepare these programs for publication in your magazine...

Greg Susong
Custom Software
Wellington, Kans.

Portable 100's "vacation" really hurt my fledgling business. What timing!

Readers deserve to hear about (products for Tandy laptops), and I deserve to be eating regularly.

I have many ideas for programs and articles, if I can survive long enough to produce them...

Mike Nugent
Tri-Mike Network East
Monroe, Mich.

The above are just a few samples of the mail we've been receiving since announcing our purchase of Portable 100. *The magazine has always been supported by a loyal cadre of readers and advertisers, and we hope our efforts at reviving the magazine can continue this tradition. As always,* Portable 100 *welcomes news and articles about products and programs. Do you have an idea for an article? Don't be shy. Send your ideas to the Editor at 80 Elm St., Peterborough, NH 03458.*

Tandy Computers: Because there is no better value.™

The Choice Is Yours

Pick the lightweight portable that suits you—whatever your needs.

The Tandy® 102—only $499. This slim, light portable gives you complete computing power at a portable price. Its small size makes it ideal for people on the go. With five built-in programs, the Tandy 102 is ready to go to work right away. Use the 102 as a personal word processor, address/phone directory, appointment calendar, telephone auto-dialer and modem. You can even write your own programs in BASIC.

The Tandy 200—more power for only $799. It's ideal for accountants, financial planners, or anyone who works with figures. You can perform sales forecasts, budgeting, pricing, engineering calculations and more. The Tandy 200 features BASIC programming language, a built-in direct-connect modem and a larger 40 × 16 display with double-height characters for maximum legibility. Built-in Multiplan™ makes spreadsheet analysis a snap. You also get an enhanced version of the Tandy 102's word-processing program, as well as an appointment calendar, address-and-phone directory, and telephone auto dialer.

Need more storage? Add our battery-powered Portable Disk Drive 2 for only $199.95. You'll be able to store up to 200K of data on a standard 3½″ disk.

Send me a new 1988 computer catalog.

Mail To: Radio Shack
Dept. 88-A-1124
300 One Tandy Center
Fort Worth, TX 76102

Name _____
Company _____
Address _____
City _____ State _____
ZIP _____
Phone _____

Radio Shack®
The Technology Store™
A DIVISION OF TANDY CORPORATION

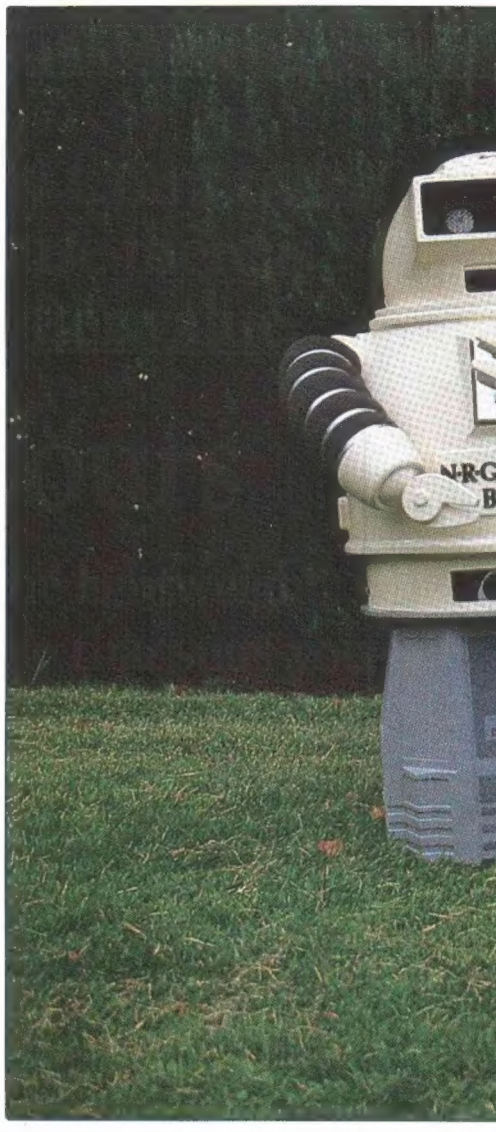
Multiplan/TM Microsoft Corp.

Circle 42 on Reader Service Card

Around the country the Model 100 is being used as a powerful, easy-to-use and inexpensive aid to learning.

Programmed to Learn

By Carl Oppedahl



Over the years, Model 100s have made their way into board rooms, warehouses and even onto oil rigs. They're used for a variety of applications from simple note taking to complex data acquisition. Now they're popping up in more logical surroundings, classrooms. And the diversity of applications is just as broad. Consider the following: Every incoming college freshman at Dallas Baptist University is equipped with a Model 100 computer, which they use in nearly every course. At Lincoln County Schools in Nevada, fifth graders type English compositions into Model 100s, and the files are transferred to a larger computer for linguistic analysis. Mount Airy City School District in North Carolina checks out Model 100s like library books, for use by students and teachers. Grinnell College, in Grinnell, Iowa, sends students off for a semester in London with Model 100 computers so they can keep in touch with on-campus advisors. Graduate students at California Polytechnic State University use Model 100s in experiments measuring human response times to various visual and audio stimuli.

At Grinnell College, often mentioned as one of the country's 10 best small liberal-arts colleges, students have been accustomed for a decade to having a large time-sharing computer for word processing and computations. But when they went on off-campus programs in Costa Rica, London or Washington, DC, they had to do without such resources. They also lost touch with faculty advisors and friends back on campus.

Then Grinnell was awarded a Sloane Foundation grant for promoting computer uses in the liberal arts. In an effort to provide computer resources to those off-campus, the college provided several Model 100s and a DEC Rainbow microcomputer to students in semester-long Washington DC internships set up through the Washington Center for Learning Alternatives. The program, supervised that year by a psychology professor, was quite successful. "It was a mind-blower," says Tom Moberg, Director of Computing at Grinnell College, "because we realized if we could do it (provide computer services) there, we could do it other places. They do everything off-campus that we do on-campus: write papers, edit papers and so on."

Of Grinnell's several off-campus programs, Grinnell-in-London is the biggest user of Model 100s. "We are sending 20 Model 100s, four printers, and a disk drive," Moberg explains. Students in the program do research in libraries, art galleries and elsewhere; the portable computers make it easy to gather information while in the field.

The Associated Colleges of the Midwest, of which Grinnell is a member, operates a Year-in-Central-America program. Each year five or six member colleges send a total of 20 to 30 students to live in private homes, most recently in Costa Rica. Grinnell provides eight Model 100s and a DEC Rainbow. Moberg explains that "students learn to write and rewrite, doing initial rough drafts on the Model 100, then uploading to the Rainbow."

Other Grinnell off-campus programs also benefit from Model 100s. A biology student posted at the Boundary Waters field station in northern Minnesota was able to use a Model 100 to speed

PHOTOGRAPHS BY GILBERT JOHNSON

The Lockheed Project at Cal Poly is a combined project where six students worked to improve the functionality of a modified HERO robot. A Model 100 is used for several purposes including remote control of motor and speech routines developed on a [illegible], downloaded to the 100, taken to the operation site, loaded into the robot and then controlled by the Model 100.

forest density calculations that take a very long time by traditional means. Anthropology students plan to use Model 100s "for field research, mostly text entry," Moberg says.

Last semester, Grinnell students in a technology assessment class wrote a study entitled "Assessment of Notebook Computer Use at Grinnell College" patterned after studies of the federal Office of Technology Assessment. They recommended college loans for student purchases of notebook computers and an increased number of college owned notebook computers available for checkout.

STANDARD EQUIPMENT

At Dallas Baptist College every incoming freshman gets a Model 100 computer, which is integrated into a majority of first-year courses, according to Professor George Poyner. "Each student turns out six or seven times as much volume of written work" as would be expected from a student with only traditional tools such as typewriter or pen and paper.

In a writing class, for example, students use the Model 100 "to help select and outline a topic." They then type in the electronic equivalent of three by five-inch note cards, then tie it all together by "bridging sentences, then cleaning up the text and making it readable."

Students in math courses use the 100 to learn about BASIC programming, structured BASIC programming, subroutines, sorting and searching. Poyner observes with pride that some of his students do quite well in programming competitions where the opponents are from high-powered, larger schools.

The university has set up a degree program in management information systems (MIS) with coursework in algorithms, hardware, networks, database management, compiler theory, and assembly language — with the Model 100 involved throughout.

At the graduate level, the Model 100 provides an ideal control unit for all sorts of laboratory work, research and solving outside problems, according to Martin Koch, acting director of the Robotics Laboratory at California Polytechnic State University in San Luis Obispo. "If you set out a piece of equipment (such as a desktop computer) it gets abuse." On the other hand, if you "check out a unit such as Model 100 to a group of students, they feel a responsibility to take care of it," he says. His school's mixed distribution of portables and desktops lets a student carry a Model 100 to the site of research and transfer the data later to the larger computer.

6 ROM BANK

plus powerful Nicad battery pack that gives 30 hours of power to your Model 100

A joint PCSG/CRYPTRONICS project.

Now you can access LUCID, DISK+, WRITE ROM, RAM+ and others instantly.

The 6 ROM BANK plus battery pack lets you have the ultimate Model 100 system. When you get your unit just press in up to 6 ROMS into the sockets then push in a simple plug into your Model 100's ROM socket. The ROM bank has two posts on either end that insert into the two little holes on the underside of your Model 100.

The ROM bank props up the Model 100 at the same angle and height as those little legs you've seen. The ROM BANK itself is only about 1½" deep and it runs the width of your Model 100. It only weighs one pound. It not only installs instantly, but it pops free in a second if you need everything to lie flat in a briefcase.

Change from ROM to ROM with the touch of a thumb switch.

You can go from LUCID to WRITE to DISK+ to any other ROMS just by turning the thumb switch at the side of the ROM bank. The 6 ROM BANK is a sturdy well built construction that looks like it is a part of your Model 100.

What is also fantastic is that the ROM bank has a powerful NICAD battery and recharger built right in. This power source supplies up to 30 hours of life to your Model 100 with just 6 hours of recharge. What's nice is that it recharges right from your Model 100's power adapter. This is a quick charge system and if you need power in a hurry, you can get 6 hours of life for your Model 100 by just charging for an hour and half.

Suddenly, the Model 100 is a very powerful computer.

If you have the HOLMES portable disk drive with its powerful bundled database software package or the CRYPTRONICS 128K RAM expansion along with LUCID, WRITE ROM and DISK+ in the 6 ROM bank with its rapid rechargable NICAD power source, you have the ultimate portable system. It's all available only from PCSG.

As usual we don't want you to take our word for it. The 6 ROM BANK is sold on a 30 day trial. If you are not satisfied simply return within thirty days for a full refund. MC, VISA, AM. EX. or C.O.D.

1-214-351-0564

PORTABLE COMPUTER SUPPORT GROUP

11035 Harry Hines Blvd. No. 207, Dallas, Tx. 75229

VISA / MASTER CARD / AM. EXP. / C.O.D.

Koch finds that Model 100s are a great help with graduate students who would usually be intimidated by computer lab equipment. His lab has Model 100s hooked up via the RS-232 port to interface boards containing Z8 microprocessors and a number of discrete inputs and outputs. With more traditional lab equipment, he finds students come to lab sessions trying to get done as soon as they can. With the user-friendliness of the Model 100s, Koch says, "You sucker them in because they enjoy it. Then they'll stay up all night, learning whatever you want them to learn."

Koch's lab actively seeks engineering projects from private industry. One work in progress is for Northern Telecom, the Canadian phone company.

Off-the-shelf Hardware: Laboratory and Factory

When you are setting up a computer system to monitor a lab experiment or to control a piece of factory equipment, there are a number of good reasons to use off-the-shelf hardware wherever possible.

Martin Koch, acting director of the Robotics Laboratory at California Polytechnic State University, gives the example of experiments in which he measures the human response time to various visual or audio stimuli. A controller that is specially made for that purpose costs tens of thousands of dollars, while a Model 100 and an inexpensive discrete input/output circuit board do the job for well under a thousand dollars.

But it's more than just savings of hardware costs. If you buy a custom-made device and it gets broken, you are stuck until the designer can be found to fix it. On the other hand, if some heavy object falls on a Model 100 and breaks it, you just unplug it and hook up another 100 in its place.

Particular advantages of the Model 100 (and Tandy 102) over custom-built controllers include the following:

- The 100 and 102 have substantial user-oriented documentation, as well as technical reference material: the Tandy service manual, my book *Inside the Model 100*, and Morgan's *Hidden Powers of the Model 100*.
- Service for the laptops is available at thousands of Radio Shack stores.
- The 100 and 102 and many popular interface boards operate on battery voltages, allowing measurement and control independent of AC power.

—C.O.

Sneaking Computers into School: A Few Tips

If you want to introduce computers into your school curriculum, consider these words of advice from educators who have already been through the process.

"Start by teaching the teachers," says David Long, director of Special Programs, Mount Airy City Schools, North Carolina. "In year one (of a two-year introduction) I would let the teachers learn to use Model 100s for lesson plans, averaging grades, word processing, making up tests and that sort of thing. In year two, go to the students."

In his school system, each building has a computer center and "the key is having a trained technician in the computer center" who knows how the computers work, can troubleshoot and can recommend software, he says.

"Go with a supplier and a system with compatibility, using portables to supplement the resources of microcomputers." suggests Martin Koch, acting director of the Robotics Laboratory, California Polytechnic State University. "You get a lot more use out of a large number of Model 100s than you would out of a smaller number of desktops," he says. "To an educator I would recommend off-the-shelf equipment rather than custom-built. Have several of each type of computer to achieve strength through redundancy — if one piece breaks you can pop in a replacement from elsewhere in the school."

Have students use portables for text entry and simple word processing, then upload to a desktop computer for final editing, printing and storage, suggests Tom Moberg, director of computing at Grinnell College. The portable is easy to teach — "we can take them in 10 minutes from never having touched a computer to the point of being able to type a document. The ROM-based software of a portable means fewer things they can screw up," Moberg says.

—C.O.

Notebook Computers: A Technology Assessment

Grinnell College (see main story) is a major user of Model 100 computers. Last semester students in an anthropology course in technology assessment studied Grinnell's use of notebook computers. The students, supervised by Professor Douglas Caulkins, patterned their study after the methods of Congress's Office of Technology Assessment.

The study opens with a general description of the technology. It classifies computer technology into three generations. The second generation, using the study's terminology, is typified by the Model 100, while the third encompasses the Data General One and other more expensive computers.

The students interviewed 18 past and present notebook computer users, including students, faculty, staff, administrators and a newspaper editor. The users identified numerous advantages and disadvantages of their notebook computer's use. Advantages included freedom from the college's mainframe computer, especially useful around mid-term week and finals week. Difficulties included inconvenient file transfers to larger computers and unfamiliar computer jargon in instructional materials.

The study found that the "notebook computer has the potential to enhance user productivity in a variety of different environments when it is used as a supplement to the Grinnell College computer facilities. Specifically, the notebook computer increases productivity when it is used for note taking during independent research and for transfer of computer mail." The increase in user productivity is attributed to the ability to "key in data at the exact point and time of reception. Such entry of data could be done while traveling, in libraries or museums or in the wilderness."

In the context of the main time-sharing computer used on campus, which becomes congested at mid-term and finals time, "students using a notebook computer would no longer have to wait for a port, contend with a very slow mainframe response time, or resort to accessing the computer at odd hours."

An appendix to the report addresses the possible problem of Model 100 keyboard noise in the college library. Team members describe an experiment in which "a team member was designated to sit and type, while the other team members observed the students in the vicinity of the typist and noted any reactions that were apparently the result of keyboard noise. After a few minutes, some of the nearby students were asked for their reactions to the noise." The group found the noise "at least slightly disturbing to all those who could hear it. Many considered the noise very annoying, while many others hardly noticed it."

The group concluded, however, that keyboard noise was not likely to be a problem over the long run. At Dallas Baptist College, where every student carries a Model 100, students have simply gotten used to the keyboards. And it is possible to silence the keys using dental rubber bands.

—C.O.

Cal Poly is writing programs to generate paper tape to control component-insertion machines. The new software replaces traditional, tedious ways of preparing the paper tape on a specialized one-character-at-a-time paper tape punch machine. One program they've written draws the patterns and creates a record of the controls required. A second program loads the information to the paper tape punch.

Koch sees Model 100s as a natural in solving such real-life engineering problems. "You achieve strength through redundancy," he says, "because if a 100 breaks, you just remove it and pop in another one." The standardized BASIC of the Model 100 also allows building up a library of commonly used routines to solve recurring needs. "You've got to build up your tool set," Koch explains.

LIBRARY BOOKS

At the other end of the educational scale, students and teachers at Mount Airy City Schools in North Carolina check out Model 100 computers much like library books. Teachers use the computers to make up tests, prepare semester-long lesson plans, calculate grades, and do word processing. Students use them for computing, for journalism and for improving typing skills.

"We don't teach a language other than BASIC, and we really go for teaching applications instead of programming," says David Long, director of Special Programs. He suggests that since real-life computer use often requires familiarity with applications rather than programming ability, it makes sense for a bookkeeping student to pick up Lucid and try at home what he's learned in class. Most of the school's 100s are equipped with PCSG's Write-ROM, he adds.

Long is particularly proud that local computer-related businesses come back and compliment the school on their graduates, and first-year college students come back and tell them it was easy because of preparation done at the high school.

Students as young as fifth grade type essays into Model 100s, which get uploaded to a Tandy 2000 programmed to analyze writing skills, according to Leo Prestwich, assistant superintendent of Lincoln County Schools in Nevada.

The Tandy 2000 measures straightforward things like correctness of spelling and capitalization, but then does more subtle analysis such as vocabulary usage and punctuation and linguistic correctness. Their 42 Model 100s are also used for telecommunications and programming classes in higher grades.

"Portables serve our purpose well," says Prestwich, "mostly because a student can type an essay for analysis without tying up an expensive desktop computer."

Tandy's introduction of the software-compatible 102 to replace the Model 100 bodes well for increased use of laptop computers in education. In the words of Grinnell College's Tom Moberg, "Use of laptop computers leaves more creative time for professors and students."□

Mini Modem

Touchbase Systems has developed the WorldLink 1200 portable modem, a battery-operated 300/1,200-bit-per-second unit designed for use with desktop and portable computers. The modem measures 4 by 2 by 1 inches and weighs 6.5 ounces with 9-volt battery. Estimated battery life is 10 hours. The modem can be directly linked into the computer or used with an acoustic cup interface for use in airports and other places where direct telephone connections are impractical.

The WorldLink 1200 has the extended Hayes AT command set, supports Bell 103/212A and CCITT v.21/v.22 protocols, features auto-dial and auto-answer, and accommodates pulse and tone dialing. Status display LEDs monitor call progress, high-speed mode, carrier detect and low battery level.

The base package's price, including battery, 7-foot phone cable and accustic cup adapter cable, is $199.

Touchbase Systems, Inc., 16 Green Acre Lane, Northport, NY 11768, (516) 261-0423.

Circle No. 90

Big Boost

The Booster Pak is a 10-ounce snap-on from Traveling Software that adds as much as 2 megabytes of RAM memory and ROM software to the Model 100 and Tandy 102. It has an optional 1,200-baud modem and rechargeable battery pack.

The basic Booster Pak includes 136K of RAM (96K available for file storage), and 64K of built-in ROM software, including TS-DOS, XMODEM communications software, and a game. The pack also has 11 open slots for RAM or ROM chips, and six more slots for 256K RAM packs. The pack adds about three-quarters of an inch to the computer's depth.

The basic Booster Pak is priced at $429. Options include 32K low-power RAM chips ($20 each), a 6-slot expansion board ($69), 256K RAM expansion modules ($159), an internal 1,200-baud Hayes-compatible modem ($200), an internal rechargeable battery pack ($69), the Desk-Link file exhange program ($40), and a four-chip set with the company's new Sardine word processor and spelling checker ($200).

Traveling Software, 19310 North Creek Parkway, Bothell, WA 98011, (206) 483-8088.

Circle No. 93

High on BASIC

Hi-Bas is a program development system from Information Systems Co. designed to reduce dramatically the time needed to develop and maintain programs for the Model 100 and Tandy 102.

Features include: Existing BASIC

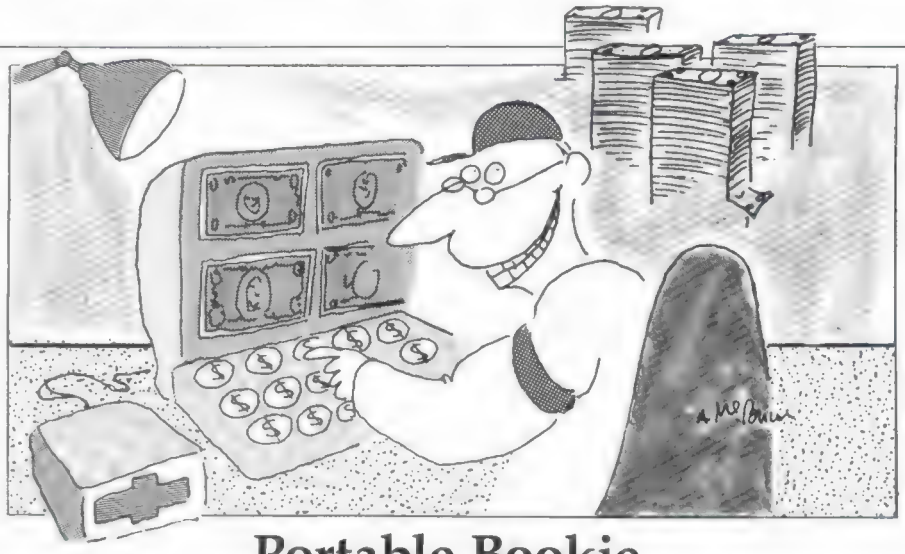
commands can be used without entering line numbers; blocks of instructions can be moved around without the need to renumber or squeeze in-between line numbers; program composing and editing is done with the TEXT full-screen editor.

Hi-Bas also extends existing BASIC with high-level commands, including structures like IFF [ELSE] ENDIF, DO CASE, and WHILE LOOP EXIT WEND. Existing BASIC commands, including GOTO, are retained. Standard structures can be nested 20 deep and intermixed. Subprograms can be copied into a program by simply calling them by name with a GOSUB. For an existing BASIC program, Hi-Bas provides an automatic conversion program that removes line numbers and creates and inserts labels.

Hi-Bas is designed for use with Model 100 and Tandy 102 computers with either the 100K or 200K Tandy PDD, and 32K DiskPower operating system. It is priced at $35, and supplied on diskette.

Information Systems Co., PO Box 353, Honolulu, HI 96809, (808) 944-8651.

Circle No. 92



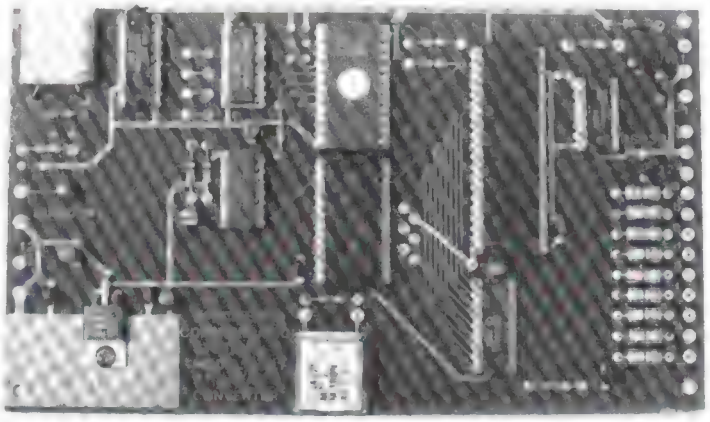
Portable Bookie

A couple of new programs from the Softtech Group may make your day at the races a little more enjoyable.

The Thoroughbred and Harness Racing Forecast programs were designed for evaluating the performance of thoroughbred and harness race horses. Data is obtained from the Daily Racing Form. Your computer performs the analysis, outputs ratings, sorts and orders the favorites. Sorted ratings may be printed on any line printer.

The Wager Return Analysis program is designed to track win, place and show finishes and percentage gains. The programs are available together as Softtech's Enhanced Horseracing Handicapping package for $49.95 plus $2 for shipping. Contact The Softtech Group Inc., P.O. Box 582, Keego Harbor, MI 48033 (313) 851-4925.

Circle No. 95



Tapping In

Connect your Model 100 to your telecommunications or remote control system or just to a telephone to monitor outgoing calls with the DTMF Decoder/Converter from Specialized Communications Systems.

The SCS Model 8501 converts all 16 standard DTMF (touch tones) to display and serial formats. The display can be any single segment common anode LED or incandescent display; any color or size. The serial output (RS-232) can be interfaced with a computer, printer or terminal. The 8501 can be connected to any audio source: speaker, discriminator, telephone coupler, etc.

Applications include control system and autopatch testing and monitoring, DTMF to computer interface and remote status indications.

Complete plans for the PC board and EPROM sell for $35; a complete kit of all parts including plans is available for $85; and a complete and fully tested board is available for $15. Contact Specialized Communications Systems, P.O. Box 16200-127, Mesa, AZ 85201.

Circle No. 94

Two from Tri-Mike

Tri-Mike Network East has developed a Dvorak keyboard system for the Model 100, Tandy 102 and 200, and NEC 8201. Available on cassette for $24.95, the keyboard is activated by one command. A transparent keyboard filter taking up fewer than 500 bytes stays out of the way, needs no further attention, and is compatible with machine-language programs.

Along a different path lies Tri-Mike's Let's Play Monopoly game for Model 100, Tandy 102, and NEC 8201 computers with at least 24K memory available. A laptop computer version of the famous Parker Brothers' game, this $29.95 package matches the user against the computer. Machine-language graphics display the entire board at all times. Players can buy, sell, mortgage and unmortgage property, build houses and hotels, or trade property with the computer. Tri-Mike warns that the computer is every bit as stingy as your little sister used to be. Uncompleted games can be saved.

Tri-Mike Network East, 33 Virginia, #22, Monroe, MI 48161, (313) 242-9693.

Circle No. 91

SOUNDSIGHT ANNOUNCES

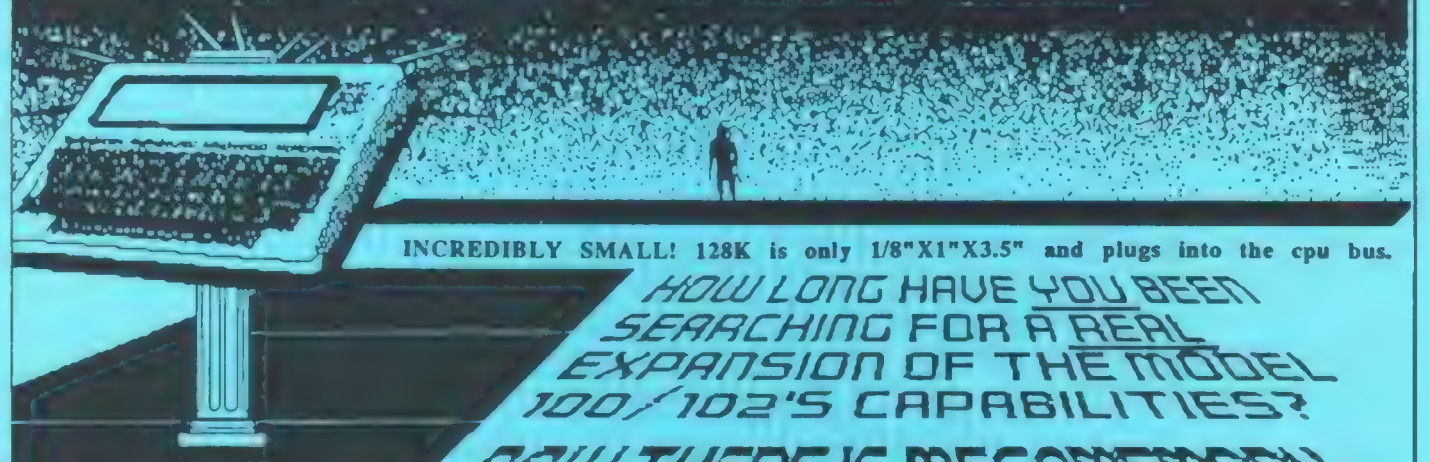
MEGAMEMORY

A NEW ERA IN RAM.

128K TO 10 MEGABYTES

RAM EXPANSION FOR THE TANDY 100 AND 102 COMPUTERS

CREATES AND EDITS* FILES AS LARGE AS THE AMOUNT OF RAM PURCHASED. FULL I O FUNCTIONS IN BASIC. FASTER, IN SOME APPS, THAN THE MODEL 100/102. OPTIONAL 4 PLACE ROM BANK. PARALLEL CPU BUS. TOTAL SIZE OF 1 MEGABYTE MODULE IS 1"X1.3"X3.5". DOES NOT COVER COMPUTER'S I O PORTS OR BATTERY COMPARTMENTS.



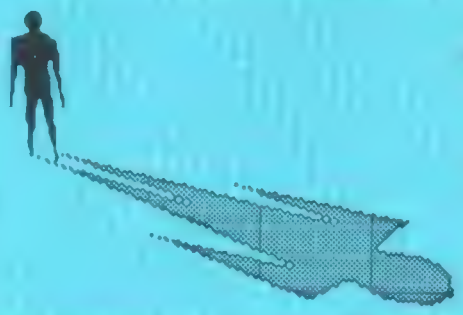
INCREDIBLY SMALL! 128K is only 1/8"X1"X3.5" and plugs into the cpu bus.

HOW LONG HAVE YOU BEEN SEARCHING FOR A REAL EXPANSION OF THE MODEL 100/102'S CAPABILITIES?

NOW THERE IS MEGAMEMORY.

Utilizing the same proven operating system software SoundSight placed in Standard Oil Service Stations from Alaska to Florida, MEGAMEMORY now offers FULL TEXT EDITING of files of any size existing in the RAM bank. *Text Editor Priced at $75.00

Currently undergoing tests for use by the LA Co. Sheriff's Dept, Megamemory also offers 3rd Party Development,(C language), POS systems and Vertical Market Applications.



The advantage of our widely used operating system lies not only in providing the first Text Editor accomodating model 100/102 files greater than 32k, but in our coming support software which includes the Text Editor, Spreadsheet, and Database.

PRICES: 128K-$329.00, 256K-$529.00, 384K-$625.00, 512K-$699.00, 640K-$769.00, 768K-$839.00, 896K-$899.00, 1 MEGABYTE-$965.00.

Additional memory prices available up to 10 megabytes in bottom mounted RAM pan. All RAM bank modules are actual byte sizes listed and do not include the Model 100 or 102's RAM.

CONTACT:

SOUNDSIGHT MBM INC. 225 W. Broadway, Suite 509, Glendale, CA., 91204 Ph (818) 240-8400

Circle 76 on Reader Service Card

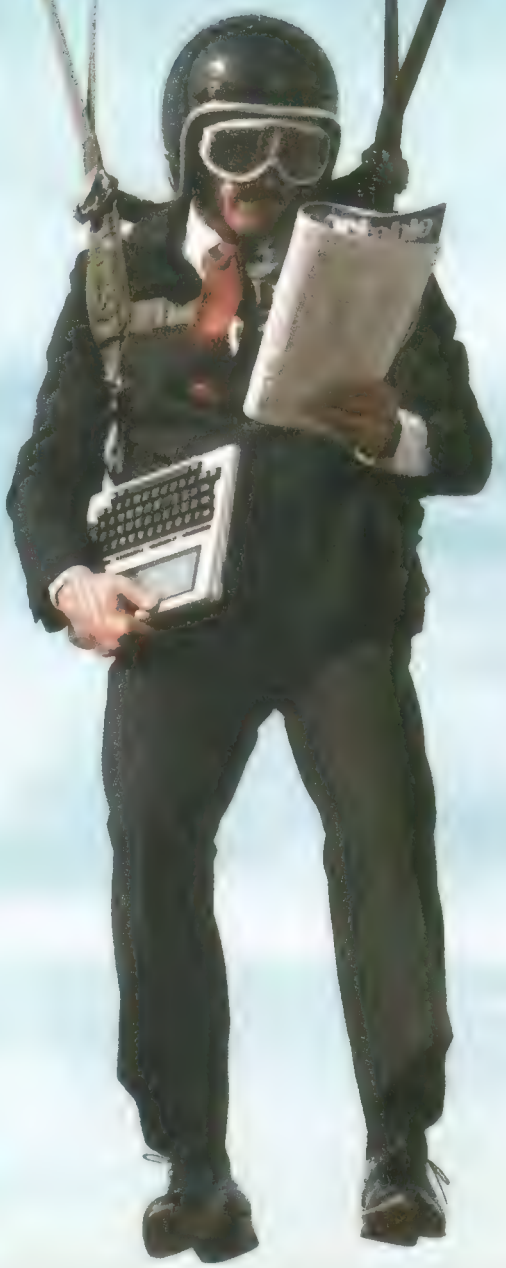
The sky's the limit...
...with Portable 100 magazine.

On the road. . . at home . . . in the office. . . in the air. . . the variety of tasks you can perform with your Tandy portable is expanding constantly.

To keep fully informed and up-to-date about current trends, new products, and new uses for your laptop, you need **Portable 100** magazine.

From sophisticated input/output (I/O) calls and their applications, to simple disk drives, **Portable 100** magazine covers it all.

Portable 100 gives you features, news, columns, and reviews that are thorough and timely. And they are written in a fast-paced, easy-to-read style, by leading experts in the computer field.



In upcoming issues you will discover. . .

- how new peripherals can make you more productive.
- more efficient ways to communicate with your desktop.
- where to buy low-cost public-domain software.
- how your business can be more profitable with your portable at your fingertips.
- where to find the best bulletin boards and information services.
- and much more!

Don't miss even a single valuable monthly issue. Fill out the coupon below, or to charge it to your credit card, call toll-free

1-800-258-5473

In N.H. call 1-924-9471.

Send no money now! We will gladly bill you.

☐ **YES!** I want to explore ways to be more productive with my Tandy portable. . . and save 47% off the newsstand price. Send me a year's subscription (12 issues) of **Portable 100** for $24.97.

☐ Payment enclosed ☐ Bill me
Make checks payable to **Portable 100**

Name _____

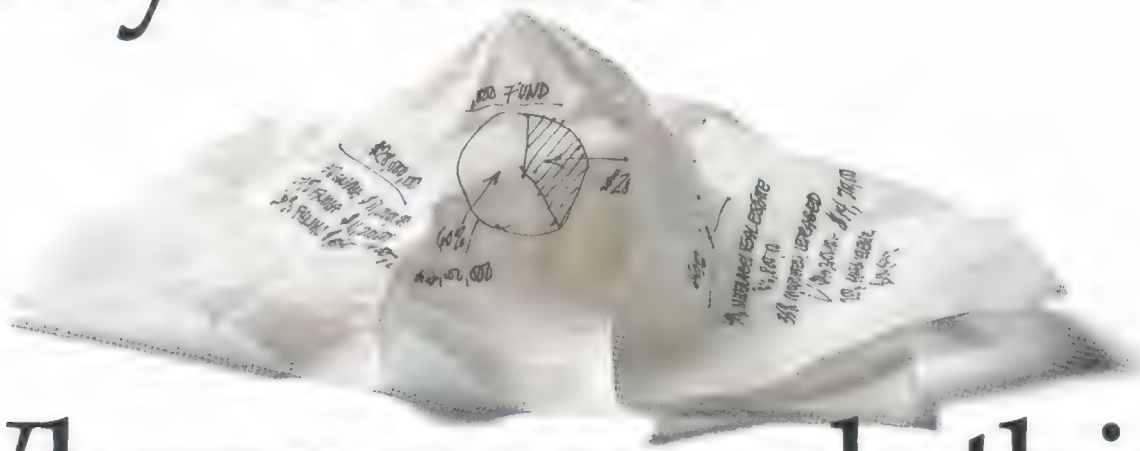
Address _____

City _____ State ____ Zip _____

Canada $45.97 (Canadian funds), Mexico, $29.97. Foreign Surface $44.97. One year only. U.S. Funds drawn on U.S. banks. Foreign Airmail, please inquire. Please allow 6-8 weeks for delivery.

MONEY BACK GUARANTEE:
If you are not completely satisfied with **Portable 100**, you may cancel your subscription and receive a full refund. Please allow 6-8 weeks for delivery of your first issue.

Why do this...



When you can do this?

Powerful portable computers can revolutionize the way you do business outside the office environment.

And ***Portable Computer Review*** is the definitive reference source for choosing the portable computer to best suit your needs.

Laptops, transportables, and hand-helds all receive in-depth scrutiny in ***Portable Computer Review.*** In addition, the Buyers' Guide is the most complete summary of all the features of all the models on the market.

The substantial range of topics includes...

- what to look for—and what to avoid—when choosing a screen
- how much memory is "enough"
- when is weight important...when is it not
- is compatibility critical
- what to check when testing the keyboard
- how to communicate with your office computer
- how much they cost...and much more!

Make sure you consult ***Portable Computer Review*** before making your purchase decision! Supplies are limited. Order your copy from **CW Communications**—the world's largest publisher of computer magazines—TODAY! For immediate service call toll free 1-800-258-5473.

Yes! I want to make the right decision. Please send me _____ copy(ies) of ***Portable Computer Review*** at $3.95 each.

☐ Payment Enclosed
☐ VISA ☐ MasterCard
☐ American Express

CARD #

EXP. DATE

SIGNATURE

NAME

STREET ADDRESS

CITY STATE ZIP

TELEPHONE

3707CD

Please make checks payable to ***Portable Computer Review.***
Delivery will begin mid-March, 1987.

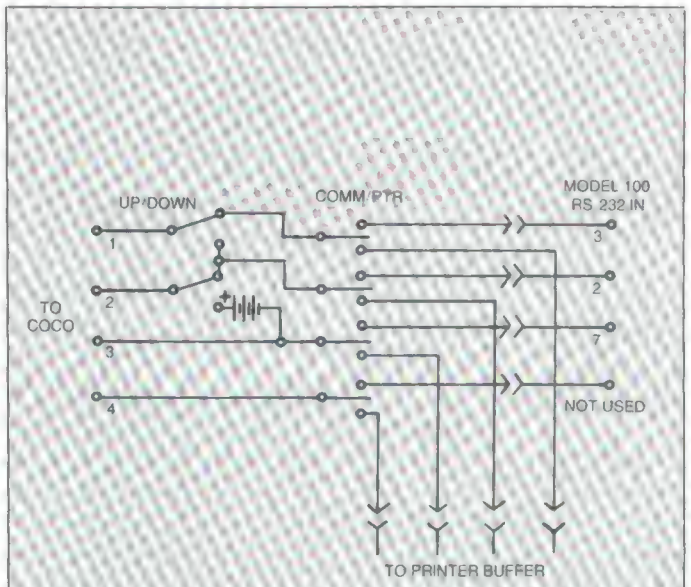
Mail to: ***Portable Computer Review,*** CW Communications/Peterborough
80 Elm Street, Peterborough, N.H. 03458

Flick That Switch

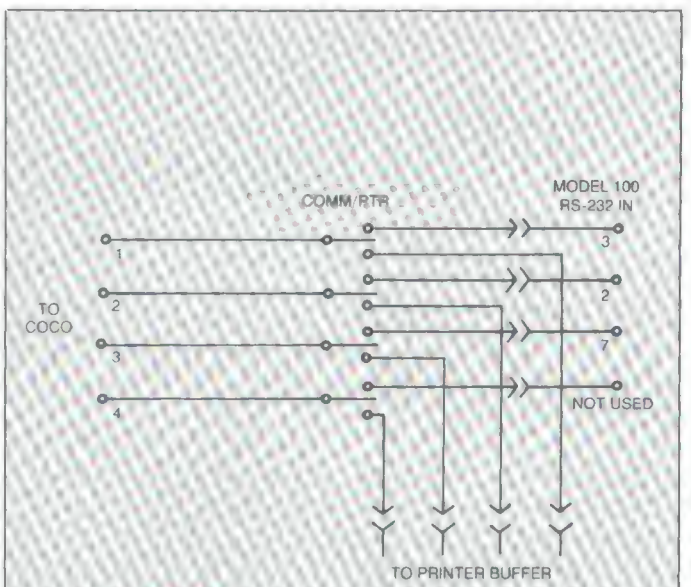
By Ralph Tenny

Soon after I got a Model 100 laptop computer, I found myself wanting to transfer material to my Color Computer (CoCo) for final polishing with Elite Word. I built a simple switchbox that allowed me to upload Model 100 files without unplugging the printer connection. With TELCOM on the Model 100 and AUTOTERM on CoCo, it all worked beautifully. But then I wanted to download to the Model 100, and nothing happened.

There were two problems involved: lack of a ready signal, and a signal conflict. RS-232 standards define two kinds of circuits: computer output and peripheral output. The DB-25 connector used on the Model 100 have the typical arrangement of output on pin 2 and input on pin 3. The standard peripherals is the oppo-



Revised circuit for UP/DOWN transfer capability.



Schematic for the original Comm/Printer switcher.

Parts List

Chassis	one required	Radio Shack #270-239
Connectors:		
Box to CoCo	one required	Radio Shack #274-018
Output	two required	Radio Shack #274-007
Switches:		
Comm/Printer	one required	Radio Shack #275-652
Up/Down	one required	Radio Shack #275-407 (slide)
		Radio Shack #275-663 (toggle)
Cables	three required	Radio Shack #26-3020

site: input on pin 2 and output on pin 3. The normal computer output circuit has direct connect cabling, with a required pin wap between pins 2 and 3 when two computers are connected for data interchange. Most computers use additional connections to pass ready signals to control data communications. The other connections are Data Terminal Ready (DTR) on the computer and Data Set Ready (DST) on the terminal. The Model 100 has the extra connections, but CoCo doesn't. (When TELCOM and AUTOTERM communicate at 300 bps, there is no timing problem and no need for the ready signal for the upload operation.)

My alteration made the data line swap and gave CoCo a ready signal at the same time. The solution was to add a second switch and battery to give a permanent ready signal. The second switch does two things: it performs the pin swap, and connects the battery to CoCo's pin 1 to make a ready signal. Note that when the new switch is in Upload, the battery is not connected to anything. Therefore, a separate power switch isn't needed if you remember to leave the unit in Upload when a session is finished.

My unit was built in a small box that was just right for the first version. Rather than start over, I just tacked on the new circuitry. The basic unit has the battery mounted on top, with the Up/Down switch inset into the side.

When the unit was first built, I had two cables; one going to the printer buffer and one to the Model 100. It was easier to drill small round holes for rubber grommets than to mount chassis-style connectors, so I ran two short cables out of the box and used cable-mount connectors to fit the existing cables. A third cable goes to the serial port on CoCo, also with a cable-mount connector.

The parts list (left) shows Radio Shack ordering numbers for all the connectors, as well as the slide and toggle switches. I had a small slide switch suitable for the Up/Down function, but it was a lot of trouble to make a rectangular cutout for the switch handle; the toggle is easier to mount. The chassis part number shown is for a larger case that will let everything fit inside. Radio Shack does not sell raw cable of the right kind, but the list shows a low-cost printer cable that has the correct connectors on one end, minimizing assembly time.

This same circuitry can be used with other computers simply by making the proper connections to a connector that fits the serial port on the new computer. Any terminal program that will accept text sent by TELCOM and create a disk file on the host computer is satisfactory.□

The Scripy Text Processor for the 100

A small but powerful solution for the Model 100 and 102.

by Thomas L. Quindry

Scripy is a machine language program that sends formatted printer output from Model 100 and Tandy 102 files. Scripy is not to be confused with a word processor. You do not write text with Scripy, you only print it. To write text, use the built-in TEXT program on the Model 100 and 102.

Though Scripy is considered a minimal function text processor, it does offer some powerful options. It takes only 320 bytes of user memory to run, whereas most commercial text processing programs for these computers use 2,000 to 3,000 bytes of valuable user memory.

THE SCRIPY PROGRAM

The Scripy text processor is listed as two BASIC programs. The main program is SCRIPY.BA. This is a BASIC file which POKES the machine language program into memory and then saves it as SCRIPY.CO. After SCRIPY.CO has been saved, SCRIPY.BA is no longer needed and can be killed. CFIG.BA is used to initially configure SCRIPY.CO. After SCRIPY.CO has been created and saved, run CFIG.BA and answer the questions concerning left margin, line length, etc. At your option, CFIG will then either save the new configuration within SCRIPY.CO or else run SCRIPY.CO with a one time configuration change.

Unless you have changing text processing needs, CFIG.BA can then be removed. The only file you need is SCRIPY.CO. Listings 1 and 2 give the SCRIPY.BA and CFIG.BA programs. The assembly code for SCRIPY.CO is given in Listing 3. Z80 code is inherently more popular among those who also have other TRS-80 computers and is listed thusly.

Listing 1. SCRIPY.BA. A BASIC program that creates SCRIPY.CO

```
10 CLS:CLEAR 50,62640
20 PRINT:PRINT TAB(11) "Loading SCRIPY.CO"
40 PRINT TAB(19) "By"
50 PRINT TAB(11) "Thomas L. Quindry":PRINT
60 FOR N=62640 TO 62959:READ A:POKE N,A:CK=CK+A:NEXT
70 READ A:IF CK<>A THEN PRINT" Checksum error, check DATA
   statements":END
80 PRINT TAB(8) "SCRIPY.CO has been saved"
90 PRINT TAB(8) "SCRIPY.BA can be killed"
100 PRINT TAB(4) "Himem must be protected to 62640"
110 SAVEM "SCRIPY.CO",62640,62959,62662
1000 DATA 10, 9, 9, 69, 110, 116, 101, 114, 0, 33
1010 DATA 176, 244, 205, 177, 39, 205, 62, 70, 218, 0
1020 DATA 0, 201, 33, 192, 245, 195, 207, 244, 33, 229
1030 DATA 245, 205, 188, 244, 33, 133, 246, 229, 14, 7
1040 DATA 205, 232, 15, 183, 202, 234, 244, 119, 254, 46
1050 DATA 202, 234, 244, 35, 13, 194, 216, 244, 6, 4
1060 DATA 17, 236, 245, 205, 105, 52, 209, 62, 9, 205
1070 DATA 171, 90, 202, 204, 244, 205, 227, 90, 229, 17
1080 DATA 133, 246, 62, 66, 18, 19, 175, 18, 14, 62
1090 DATA 225, 17, 135, 246, 13, 202, 66, 245, 126, 6
1100 DATA 1, 254, 136, 202, 39, 245, 4, 254, 156, 202
1110 DATA 39, 245, 6, 3, 254, 157, 194, 45, 245, 120
1120 DATA 50, 147, 245, 62, 32, 18, 35, 19, 254, 140
1130 DATA 202, 90, 245, 254, 26, 202, 89, 245, 254, 13
1140 DATA 194, 14, 245, 195, 89, 245, 229, 213, 43, 27
1150 DATA 126, 183, 194, 82, 245, 209, 225, 43, 27, 195
1160 DATA 90, 245, 254, 32, 194, 68, 245, 193, 193, 35
1170 DATA 229, 62, 13, 18, 14, 10, 62, 32, 205, 184
1180 DATA 245, 13, 194, 96, 245, 33, 135, 246, 126, 254
1190 DATA 140, 202, 131, 245, 254, 26, 202, 140, 245, 254
1200 DATA 13, 202, 146, 245, 205, 184, 245, 35, 195, 108
1210 DATA 245, 205, 160, 245, 205, 185, 244, 195, 255, 244
1220 DATA 205, 160, 245, 195, 0, 0, 62, 1, 205, 163
1230 DATA 245, 126, 254, 13, 218, 131, 245, 195, 8, 245
1240 DATA 58, 133, 246, 245, 33, 133, 246, 53, 62, 13
1250 DATA 205, 184, 245, 62, 10, 205, 184, 245, 241, 61
1260 DATA 200, 195, 163, 245, 245, 230, 127, 205, 63, 109
1270 DATA 241, 201, 10, 9, 9, 83, 99, 114, 105, 112
1280 DATA 121, 10, 13, 10, 9, 32, 98, 121, 32, 84
1290 DATA 104, 111, 109, 97, 115, 32, 76, 46, 32, 81
1300 DATA 117, 105, 110, 100, 114, 121, 10, 10, 13, 9
1310 DATA 9, 70, 105, 108, 101, 0, 46, 68, 79, 0, 42518
```

FEATURES

Scripy offers some powerful features. All default settings can be changed using CFIG.BA. Current features and their default settings are:

1. Set skip over perforation - Default 12 lines.
2. Set left margin - Default 10 spaces.
3. Set maximum line length - Default 60 characters.
4. Set linefeed - Default 1 line.
5. Within text selectable/changeable linefeeds (1, 2, or user defined with GRPH codes).
6. Optional linefeed with carriage return.
7. Within text embedding of control codes to enable printing features for any brand printer.
8. Optional pause after formfeed.
9. Cold start default values can be changed by the user.
10. Exit program at any time by hitting the SHIFT-BREAK key combination.

Scripy can be used by practically any parallel printer. Special printer functions can be attained by embedding control codes within text as explained in narrative that follows.

The GRPH key is used to send control codes to the printer from within text. Three codes are reserved for changing the number of linefeeds sent with each carriage return. The default linefeed when running Scripy can be changed within the text file by substituting a special graphic character for a space character.

Thus, text files can have different linespacings within the same text as needed. GRPH-1, GRPH-2, or GRPH-3 (depending on whether 1, 2, or 3 linefeeds are desired) is substituted for a space in the text. (The GRPH 3 code can be set to any desired linespacing by CFIG.BA.) Text printing starting with the next line after substitution will use the new linefeed specification.

This instruction as well as other instructions to Scripy can be changed within text as many times as desired. If a linefeed default other than 1 is desired, a good way to change the default is to save the SCRIPY.CO file after printing a text file using the number of linefeeds desired for the default. The file can be saved using the command, SAVEM "SCRIPY.CO", 62640, 62959, 62662.

An easier way is to use the small BASIC program called CFIG.BA to change Scripy default values and save the modified file. You can also use the POKE values found in CFIG.BA to change individual parameters. Scripy uses the keyboard input buffer to store the line of text to be LPRINTed next. This buffer is 254 bytes long.

Since Scripy uses two of these bytes for a carriage return character and overhead, line lengths greater than 252 bytes are not recommended and may give unpredictable results. Once you have set the default values, CFIG.BA saves the modified file as described above or runs Scripy with the changes being only temporary.

CONTROL CODES

Scripy can send any control from 0 to 127 to the printer. This is done using the GRPH key on the Model 100. Add 128 to any code you care to send. Consult your printer manual for the proper code and your Model 100 manual for the proper GRPH key code to embed it in your text. Scripy filters the seventh bit, which allows you to send these codes. Do not substitute these codes for a space as you do with the linefeed control codes. Each control code is counted as a character and sent to the printer. The only other exception is the code to force a formfeed.

GRPH ' is substituted for the <CR> symbol. Table 1 gives several examples of control codes that can be sent with the GRPH key. The descriptions of functions performed are Epson-specific with the exception of the first three.

Graphic codes similar to those given in Table 1 can be grouped to give any of the ESC code combinations by the printer. With an Epson printer, for example, "GRP H k", "E" will turn on the emphasized mode. "GRPH k", "F" will turn it off. "GRPH k, "-", " GRPH m" will turn on underlining and "GRPH k, "-", " GRPH p" will turn it off.

PROTECTING MEMORY

High memory must be protected by going into BASIC and setting the HIMEM pointer using the command, CLEAR 50,62640. SCRIPY.CO can then be loaded and run from the menu. Running the BASIC program to create SCRIPY.CO protects this high memory automatically so you don't have to repeat it unless you later change your HIMEM pointer.

TECHNICAL NOTES

To aid other programmers who may want to utilize some of the ROM calls and other vectors used in this program, Table 2 summarizes those aspects of ROM and vectors used by the operating system.

For a copy of this program and also LENGTH.CO from this issue, send $6 for a 3.5-inch disk for the Tandy Portable Disk Drive to Thomas L. Quindry at 6237 Windward Drive, Burke, VA 22015.

Note: The version of Scripy described is not intended to work with the Radio Shack Model 100 Disk Video Interface (DVI) System. A version for the DVI is included on the above mentioned program disk. These programs have previously been placed into the Public Domain.

Scripy has no affiliation with any version of the Scripsit word processor programs distributed by Radio Shack. □

Listing 2. CFIG.BA: A BASIC program to provide initial configuration to SCRIPY.CO.

```
10 CLS:LOADM "SCRIPY"
20 PRINT, "Scripy Config":PRINT
30 A=10:INPUT "Left Margin (10)";A:POKE 62815,A
40 A=60:INPUT "Line length (60)";A:A=A+2:POKE 62729,A
50 A=1:INPUT "Default linefeeds (1)";A:POKE 62867,A
60 A=3:INPUT "No. linefeeds for GRPH-3 (3)";A:POKE 62753,A
70 A=12:INPUT "Skip over perforation (12)";A:A=A+1:POKE62873,A
80 A=66:INPUT "Page length (66)";A:POKE 62723,A
90 PRINT "Pause after formfeed? ";:GOSUB 1000:IF A=1 THEN POKE
   62649,33 ELSE POKE 62649,201
100 PRINT "Linefeed with carriage return? ";:GOSUB 1000:IF A=1
    THEN POKE 62896,184 ELSE POKE 62896,191
110 PRINT "Save Configuration? ";:GOSUB 1000:IF A=1 THEN PRINT
"Saved": SAVEM "SCRIPY",62640,62959,62662 ELSE CLS: CALL 62662
999 END
1000 A=ASC(INPUT$(1)) AND 223:IF A=89 THEN A=1: PRINT "Yes":
     RETURN ELSE IF A=78 THEN PRINT "No":RETURN ELSE GOTO 1000
```

Listing 3. Z80 Assembly code for SCRIPY.CO.

```
00100 ;*********************************
00110 ;*                               *
00120 ;*            Scripy             *
00130 ;*              by               *
00140 ;*       Thomas L. Quindry       *
00150 ;*     6237 Windward Drive       *
00160 ;*       Burke, VA 22015         *
00170 ;*                               *
00180 ;*  Version of October 29, 1985  *
00190 ;*                               *
00200 ;*********************************
00210 ;
00220          ORG     62640           ;START AT 62662, F4C6H
00230 MESS4    DEFB    0AH
00240          DEFW    0909H
00250          DEFM    'Enter'
00260          DEFB    0
00270 PAUSE    LD      HL,MESS4        ;PAUSE MESSAGE
00280 DISPL1   CALL    27B1H           ;DISPLAY STRING
00290          CALL    463EH           ;GET <ENTER> AND RETURN
00300          JP      C,0000          ;MENU IF BREAK
00310          RET
00320 START    LD      HL,MESS1        ;TITLE
00330          JR      DISPLA
00340 NOFIND   LD      HL,MESS3        ;REPEAT IF BAD FILENAME
00350 DISPLA   CALL    DISPL1          ;DISPLAY STRING
00360          LD      HL,0F685H       ;INPUT BUFFER
00370          PUSH    HL
00380          LD      C,7             ;CHECK FILENAME
00390 CHAR     CALL    0FE8H           ;CHECK UPPER/LOWER CASE
00400          OR      A
00410          JP      Z,ZERO          ;CHECK FOR END OF INPUT
00420          LD      (HL),A
00430 CHAR1    CP      '.'
00440          JP      Z,ZERO          ;CHECK FOR PERIOD
00450          INC     HL
00460          DEC     C
00470          JP      NZ,CHAR         ;ONLY LOOK FOR 7 CHAR
00480 ZERO     LD      B,4
00490          LD      DE,MESS2        ;MAKE EXTENSION A .DO
00500          CALL    3469H           ;MOVE FROM (DE) TO (HL)
00510          POP     DE              ;GET BUFFER ADDRESS
00520          LD      A,9             ;SET TO 9 CHARACTERS MAX
00530          CALL    5AABH           ;FIND DIRECTORY ENTRY
00540          JP      Z,NOFIND        ;IF CAN'T FIND IT
00550          CALL    5AE3H           ;GET FILE ADDRESS
00560          PUSH    HL              ;SAVE FILE ADDRESS
00570 PL       LD      DE,0F685H       ;SET UP PAGE LENGTH FLAG
00580          LD      A,66            ;LINES PER PAGE
00590          LD      (DE),A          ;STORE IT
00600          INC     DE              ;SET UP PRINT BUFFER
00610          XOR     A
00620          LD      (DE),A          ;ZERO AT FRONT OF BUFFER
00630 NEXT     LD      C,62            ;LINE LENGTH + 2
00640          POP     HL              ;CURRENT FILE POINTER
00650          LD      DE,0F687H       ;BEGINNING OF TEXT BUFFER
00660 TEST     DEC     C
00670          JP      Z,SPCCHK        ;GO IF MAX LINE LENGTH
00680          LD      A,(HL)          ;GET TEXT CHARACTER
00690          LD      B,1
00700          CP      88H             ;CHECK FOR GRPH-1
00710          JP      Z,CHG
00720          INC     B
00730          CP      9CH             ;CHECK FOR GRPH-2
00740          JP      Z,CHG
00750          LD      B,3             ;USER SPECIFIED
00760          CP      9DH             ;CHECK FOR GRPH-3
00770          JP      NZ,NOCHG
00780 CHG      LD      A,B
00790 LINES    LD      (NEXT1+1),A     ;SET LINESPACE COUNTER
00800          LD      A,20H           ;REPLACE WITH SPACE CHAR
00810 NOCHG    LD      (DE),A          ;LOAD CHAR IN BUFFER
00820          INC     HL              ;POSITION FOR NEXT CHAR
00830          INC     DE              ;POSITION FOR NEXT CHAR
00840          CP      140             ;FORM FEED COMMAND?
00850          JP      Z,LPRIN2
00860          CP      1AH             ;CHECK FOR EOF
00870          JP      Z,CR
00880          CP      0DH             ;CHECK FOR <CR>
00890          JP      NZ,TEST
00900          JP      CR              ;<CR> ENDS LINE
00910 SPCCHK   PUSH    HL              ;SAVE CURRENT LINE PTR
00920          PUSH    DE              ;SAVE CURRENT BUFFER PTR
00930 LPRIN1   DEC     HL              ;CHECK FOR LAST SPACE
00940          DEC     DE
00950          LD      A,(HL)
00960          OR      A
00970          JP      NZ,SPACE        ;IF NOT AT BUFFER BEGIN
00980          POP     DE              ;RESTORE BUFFER PTR
00990          POP     HL              ;RESTORE LINE PTR
01000          DEC     HL
01010          DEC     DE
01020          JP      LPRIN2
01030 SPACE    CP      ' '
01040          JP      NZ,LPRIN1       ;GO BACK IF NOT A SPACE
01050          POP     BC              ;JUSTIFY STACK
01060          POP     BC              ;JUSTIFY STACK
01070 CR       INC     HL
01080 LPRIN2   PUSH    HL              ;CURRENT LINE POINTER
01090          LD      A,0DH           ;PUT <CR>
01100          LD      (DE),A
01110 LPRIN3   LD      C,10            ;LEFT MARGIN SPACES
01120 SPACES   LD      A,' '           ;PRINT SPACES
01130          CALL    LPRINT
01140          DEC     C
01150          JP      NZ,SPACES       ;REPEAT FOR NO. OF SPACES
01160          LD      HL,0F687H       ;BEGINNING OF TEXT BUFFER
01170 LPRIN4   LD      A,(HL)          ;GET CHARACTER
01180          CP      140             ;FORM FEED COMMAND?
01190          JP      Z,FORMS
01200          CP      1AH             ;LOOK FOR EOF
01210          JP      Z,ENDER         ;IF EOF DO FORMFEED
01220          CP      0DH             ;LOOK FOR <CR>
01230          JP      Z,NEXT1
01240          CALL    LPRINT
01250          INC     HL
01260          JP      LPRIN4          ;GO BACK FOR NEXT CHAR
01270 FORMS    CALL    FF              ;DO A FORMFEED
01280          CALL    PAUSE           ;PAGE PAUSE
01290          JP      PL              ;RESET PAGE LENGTH COUNT
01300 ENDER    CALL    FF              ;DO A FORMFEED
01310          JP      0000            ;PROGRAM END
01320 NEXT1    LD      A,1             ;NUMBER OF LINEFEEDS
01330          CALL    LF              ;DO LINEFEEDS
01340          LD      A,(HL)
01350          CP      13              ;SKIP OVER LINEFEEDS +1
01360          JP      C,FORMS         ;DO FORMFEED IF NEEDED
01370          JP      NEXT
01380 FF       LD      A,(0F685H)      ;GET LINECOUNT
01390 LF       PUSH    AF
01400          LD      HL,0F685H
01410          DEC     (HL)            ;DECREMENT LINECOUNT
01420          LD      A,0DH           ;DO A <CR>
01430          CALL    LPRINT
01440          LD      A,0AH           ;DO A LINEFEED
01450          CALL    LPRINT
01460          POP     AF
01470          DEC     A
01480          RET     Z               ;RETURN IF DONE
01490          JP      LF              ;DO ANOTHER LINEFEED
01500 LPRINT   PUSH    AF              ;LPRINT SUBROUTINE
01510          AND     127             ;GET CONTROL CODES
01520          CALL    6D3FH           ;LPRINT ROUTINE
01530          POP     AF
01540          RET
01550 MESS1    DEFB    0AH             ;TITLE BLOCK
01560          DEFW    0909H           ;TABS
01570          DEFM    'Scripy'
01580          DEFB    0AH
01590          DEFW    0A0DH
01600          DEFB    09H
01610          DEFM    'by Thomas L. Quindry'
01620          DEFW    0A0AH
01630          DEFB    0DH
01640 MESS3    DEFW    0909H
01650          DEFM    'File'
01660          DEFB    0
01670 MESS2    DEFM    '.DO'           ;DOCUMENT EXTENSION NAME
01680          DEFB    0
01690          END     START
```

Now you can put your own programs on a plug-in ROM chip.

A custom ROM service designed especially for corporate and government clients.

Now, Traveling Software can transform your own BASIC or machine code programs into executable applications in a built-in ROM chip—by using a process called "Guardian" that took two years and thousands of man hours to develop.

Fast, simple, economical and guaranteed.

It's a very simple, speedy process to have your own applications "burned" into a plug-in ROM chip. All you do is send Traveling Software your programs on a disk with a check for $500 for your first ROM. Then, the Traveling Software engineers will convert your program(s) into a plug-in ROM chip.

Quantity purchases can achieve a per ROM cost of only $49, with full credit for the initial $500 cost. Normal service turnaround is 7-10 working days. Optional rush service of 24-48 hours is possible.

Combine Traveling Software Programs with your own.

For orders of 50 or more custom ROM's, Traveling Software is offering the option to combine popular products like TS-DOS, T-Word, IDEA!, or T-base with your programs on a single chip.

Build your own "ultimate ROM" using any Traveling Software product with your own custom applications.

The possibilities are endless! For instance, you can even join a bar code reader data logging application with the disk access powers of TS-DOS and put it on a single plug-in chip!

Tandy and NEC Computer Sales.

Looking for a Tandy Model 100/102, 200 or NEC PC-8201 laptop computer to go along with your custom ROM's? Traveling Software is now an authorized distributor of these computers. Let Traveling Software take care of all your laptop computing needs with just one purchase order!

- Proven technology, thousands of Guardian ROM's have been installed since early 1985
- Traveling Software has years of experience with laptop computers and ROM technologies
- Economical solutions to many applications, as low as $49 per custom ROM
- Built-in future protection! Program changes can be "reburned" into existing chips without replacing costly CMOS ROM chips

A few current Guardian ROM customers:

- Associated Press
- McDonald's Corporation
- Swiss News Agency
- Matrix Instruments, Inc.
- Professional Growth Associates
- Biolabs Inc.
- Shell Pipeline Company
- Del Mar Avionics
- Business Systems Design, Inc
- Ocean State Scale
- Pruett Industries
- Specialty Instruments

[redacted]

Tom Villars
Project Manager
Enterprise Systems, Inc

CUSTOM SERVICES

*For your FREE **ROM** information kit call toll free*

1 800 343-8080

Or write **Traveling Software**
North Creek Corporate Center
19310 North Creek Parkway
Bothell, Washington 98011
206-483-8088

TRAVELING SOFTWARE

Circle 72 on Reader Service Card

GRAPHIC BY PAIGE PARKER-TAYLOR

A Font for All Seasons

By Jay Holovacs

An excellent feature of the popular Epson FX series of printers is support for user-defined typefaces in printer random access memory (RAM). Once a character set is downloaded into the printer, text can be printed in the alternate typeface. Or you can even switch back and forth between the new and the standard typeface just as if they were a part of the printer ROM.

This allows for interesting possibilities, including special typefaces (old English, extra bold, block, company logo) or exotic foreign alphabets such as Cyrillic or Hebrew.

The characters are defined in a manner similar to graphics. However, character definition is a separate capability that can be printed much faster than characters constructed individually from graphics commands. Most of the special features, such as super and subscripts, underlining, emphasized print, proportional print, double strike and pica/elite will work with user-supplied character sets.

When I first started working with this feature of the printer, I found implementing it was no easy task. It involves a tedious conversion of penciled graphics into binary codes that then need to be prepared for the actual download into the printer.

To simplify this drudgery, I began to develop FONTFX.BA, a typeface editor program. FONTFX allows for graphic editing, saving, downloading and modification of typefaces. Type data can be stored in RAM, or on cassette or disk for downloading at the start of a printing session.

FONTFX is written for the Model 100 but also runs on the Tandy 200. Users may want to modify screen placement to take advantage of the larger screen size, but this is not required.

RUNNING FONTFX

You'll first be prompted for the file name of an existing file to edit. If this is a new session, simply hit Enter.

The next screen is the command line. Hit E for edit. You'll be prompted for an ASCII value to edit, then for its characteristics (for example, whether it's a descender character such as 'y' or 'g' and where, in dot columns, the character starts and ends). To accept default characteristics, simply hit enter at each prompt (65, No, 0 and 11 are the defaults). Once these entries are completed, the character is drawn on a grid on the screen (blank grid is the default). The characteristics can be changed later, so it's good to start with the defaults when first developing a character.

Once in the Edit mode, the following choices are available as key-commands (use upper case for commands):

- EDIT returns you to the top of the edit screen; the ASCII or characteristics can then be changed.
- FILL fills the current pixel location (as indicated by the pixel cursors along the top and side of the character frame).
- CLEAR clears the current pixel.
- BLANK erases the currently selected character.
- MOVE is followed by a choice of five subcommands (Up, Down, Right Left and [enter]=abort). Entering the key command will move the entire shape one pixel in the indicated direction.
- SAVE saves the current set of character definitions as a file (can be RAM, cassette, or with proper DOS direct-to-disk).
- LOAD loads an existing character set file. Note: if you have characters in the program, they will be overwritten by this command.
- DOWN downloads the current character set to printer. You will be given the opportunity to select which ASCII values will be downloaded, and to copy the ROM characters into RAM first (this allows default to any standard characters that have not been redefined.)
- TEST prompts for a string, which then will be printed out, for the purpose of evaluating the characters on paper. You will be able to choose

FONTFX.BA, a typeface editor program for Epson FX series printers.

```
1 REM FONTFX 1.0C 6/12/86; Jay Holovacs
5 DEFINT A-Z
   :DIM CD(94, 11)
   :XCNR=190
   :YCNR=10
   :AC$=CHR$(28)+CHR$(29)+CHR$(30)+CHR$(31)+"FCSLDTKQMBE"
   :ES$=CHR$(27)
   :ZO$=CHR$(0)
   :FL=0
6 FOR CHAR=0 TO 94
   :CD(CHAR, 0)=139
   :NEXT CHAR 'DEFAULT PARAMETERS
7 ON ERROR GOTO 1600
10 CLS
   :PRINT @93, "* * FONTFX * *"
   :PRINT
   :PRINT @161, "Typeface editor for Epson FX printers"
   :PRINT @220, "  by Jay Holovacs"
   :PRINT @295, ". . hit any key";
   :R$=INPUT$(1)
20 GOSUB 1200
   :CLS
100 REM LOOP
105 GOSUB 950
110 R$=INKEY$
   :IF R$="" THEN 110
115 IF INSTR(1, AC$, R$) THEN 117 ELSE 110
117 ON INSTR(1, AC$, R$) GOTO 120, 122, 124, 125, 126, 127, 128,
    129, 130, 131, 132, 140, 133, 134, 135
120 DIR=1
   :GOSUB 700
   :GOTO 110 'RIGHT
122 DIR=-1
   :GOSUB 700
   :GOTO 110 'LEFT
124 DIR=-1
   :GOSUB 750
   :GOTO 110 'UP
125 DIR=1
   :GOSUB 750
   :GOTO 110 'DOWN
126 IF NE THEN 110 ELSE SW=1
   :GOSUB 500
   :GOSUB 800
   :FL=1
   :GOTO 110 'FILL BIT
127 IF NE THEN 110 ELSE SW=0
   :GOSUB 500
   :GOSUB 800
   :FL=1
   :GOTO 110 'CLEAR BIT
128 GOSUB 1100
   :GOTO 100 'SAVE
129 GOSUB 1200
   :GOTO 100 'LOAD
130 GOSUB 1300
   :GOTO 100 'DOWNLOAD
131 GOSUB 1400
   :GOTO 100 'TEST
132 GOSUB 1500
   :GOTO 100 'KLONE
133 GOSUB 1700
   :GOTO 110 'SHIFT CHAR
134 GOSUB 1550
   :GOTO 110 'BLANK A CHAR
135 GOSUB 900
   :GOTO 100 'EDIT MODE
```

```
140 CLOSE
   :CLS
   :IF FL THEN PRINT @40, "Data modified since last save"
   :PRINT "QUIT (Y/N)?";
   :IF INPUT$(1)<>"Y" THEN 100
142 END
300 REM DRAW GRID--ENTRY/XCNR, YCNR
320 LINE(XCNR, YCNR)-(XC+4, YC+41), 1, B
322 FOR X=5 TO 40 STEP 5
   :LINE(XC, X+YC)-(XC-3, X+YC)
   :NEXT
325 FOR X=4 TO 33 STEP 3
   :LINE(X+XC, YC)-(X+XC, YC-3)
   :NEXT
330 RETURN
400 REM DRAW PIXEL--ENTRY/LROW, LCOL, SW
405 FOR Q=0 TO 2
   :FOR Q1=0 TO 3
   :PSET(LCOL+Q, LROW+Q1, SW)
410 NEXT Q1, Q
   :RETURN
500 REM DRAW A PIXEL--ENTRY/SW, RCUR, CCUR
502 IF NE THEN RETURN
505 LROW=YCNR+1+5*(RCUR-1)
   :LCOL=XCNR+1+3*(CCUR-1)
   :GOSUB 400
   :RETURN
600 REM PLACE CURSOR MARKERS--ENTRY/SW, RCUR, CCUR=CURSOR
    LOCATION
620 LROW=YCNR-6
   :LCOL=XCNR+1+3*(CCUR-1)
   :GOSUB 650
   :RETURN 'HORIZONTAL
630 LCOL=XCNR-5
   :LROW=YCNR+1+5*(RCUR-1)
   :GOSUB 650
   :RETURN 'VERTICAL
650 REM DRAW CURSORS--ENTRY/LCOL, LROW
655 FOR Q=1 TO 2
   :FOR Q1=1 TO 2
   :PSET(LCOL+Q, LROW+Q1, SW)
660 NEXT Q1, Q
   :RETURN
700 REM LEFT/RIGHT SCROLL--ENTRY/DIR, CCUR
705 IF CCUR+DIR<1 OR CCUR+DIR>11 OR NE THEN RETURN
710 SW=0
   :GOSUB 620
   :CCUR=CCUR+DIR
   :SW=1
   :GOSUB 620
   :RETURN
750 REM UP/DOWN SCROLL--ENTRY/DIR, RCUR
755 IF RCUR+DIR<1 OR RCUR+DIR>8 OR NE THEN RETURN
760 SW=0
   :GOSUB 630
   :RCUR=RCUR+DIR
   :SW=1
   :GOSUB 630
   :RETURN
800 REM MODIFY A BIT IN ARRAY VALUE--ENTRY/SW, CHAR, RCUR, CCUR
805 IF SW THEN CD(CHAR, CCUR)=CD(CHAR, CCUR) OR 2^(8-RCUR) ELSE
    CD(CHAR, CCUR)=CD(CHAR, CCUR) AND NOT(2^(8-RCUR))
810 RETURN
900 REM ACCEPT PARAMETERS
905 NE=0
   :SW=1
   :CLS
   :RCUR=1
   :CCUR=1
   :GOSUB 620
```

standard or proportional spacing.

- KLONE copies a specified ASCII into another. This is a useful starting point for characters that are similar such as Q and O.
- QUIT exits the program. If changes have been made since last Save, you will be notified and given a chance to reconsider.

Additionally, the four arrow keys manipulate the bit cursor locations. Key commands that prompt for additional information (Save, Load etc.) can be aborted by entering a null in response to the prompts.

> You can switch back and forth between the new and standard typeface.

You should note that when loading or quitting, a warning message will appear if the character data has been modified since it was last saved, providing the opportunity to save the current definitions before losing them.

PIXEL SPACING

The FX printer places a limitation on dot placement such that dots cannot be directly adjacent right to left. For that reason, a horizontal gap of at least one pixel must remain between pixels on the same row. Pixels may be directly vertically or diagonally adjacent. Even though the white space is quite noticeable on the screen, there is considerable overlap on paper and the result is not as drastic as might be expected. It's helpful to frequently download and test characters to get an accurate feel for the procedure.

Simulating curved lines and various slopes is not really as difficult as might be imagined from looking at the large character frame. Use the download feature frequently at first to get a good feel for the techniques of character description.

Once created and saved, typeface files can be downloaded before a printing session by running FONTFX, loading the appropriate file and going directly to the download command.

After the data is downloaded to the

printer, the alternate typeface can be summoned by the sequence CHR$(27);-CHR$(37);CHR$(N);CHR$(0). If N = 0 then standard typeface is used. If N = 1 then RAM type is used.

Many word processing programs provide a means to embed control sequences directly in your text. Unfortunately, this command is not directly available in the standard text dump built into the Model 100 because it has no means to embed a CHR$(0). However, the command can be sent from BASIC, which lets you switch over to text to utilize the RAM type.

DESCENDERS

When a descender is specified in the characteristics, the pixels are entered in the same manner as usual, but the entire frame will be shifted one pixel down on paper when the character is printed.

Proportional spacing can be specified when printing with the usual embedded sequence — CHR$(27);-CHR$(112);CHR$(1) — but you must supply the proportional information to the printer when editing the typeface. The characters in the Epson FX consist

```
    :GOSUB 630
    :GOSUB 300
910 PRINT @5, "* *EDIT CHARACTER* *"
    :CHAR=65
    :INPUT "ASCII (33-127)";CHAR
    :CHAR=CHAR-33
    :IF CHAR<0 OR CHAR>94 THEN 910
911 PRINT @55, CHAR+33
912 IF CD(CHAR, 0) AND 128 THEN PRINT @97, "N"; ELSE PRINT
    @97, "Y";
913 R$=""
    :PRINT @80, ;
    :INPUT "DESCENDER (Y/N)";R$
    :IF R$<>"" THEN FL=1
    :IF R$<>"Y" THEN CD(CHAR, 0)=CD(CHAR, 0) OR 128 ELSE
    CD(CHAR, 0)=CD(CHAR, 0) AND 127
920 PRINT @137, (CD(CHAR, 0)\16) AND 7;
    :QQ=0
    :PRINT @120, ;
    :INPUT "CHAR START (0-7)";QQ
    :IF QQ THEN PRINT @137, QQ
    :QQ=(QQ AND 7)*16
    :CD(CHAR, 0)=(CD(CHAR, 0) AND 112) OR QQ
    :FL=1
930 PRINT @176, CD(CHAR, 0) AND 15;
935 QQ=12
    :PRINT @160, ;
    :INPUT "CHAR END (5-11)";QQ
    :IF QQ<>12 THEN FL=1
    :CD(CHAR, 0)=(CD(CHAR, 0) AND 240) OR QQ
    :PRINT @176, QQ 'ALLOW FOR DEFAULT ENTRY
940 GOSUB 1000
```

portable 100

Portable 100 is a member of the International Data Group, the world's largest publisher of computer-related information. The group publishes more than 80 computer publications in more than 28 countries. Fourteen million people read one or more of the group's publications each month. Members of the IDG group contribute to the *CW International News Service*, offering the latest domestic and international computer news.

Members of the group include: ARGENTINA's *Computerworld Argentina, PC Mundo;* ASIA's *Asian Computerworld;* AUSTRALIA's *Computerworld Australia, Communications World, Australian PC World, Australian Macworld;* AUSTRIA's *Computerwelt Osterreich;* BRAZIL's *DataNews, PC Mundo, Micro Mundo;* CHILE's *Informatica, Computacion Personal;* DENMARK's *Computerworld Danmark, PC World Danmark;* FINLAND's *Tietoviikko, Mikro;* FRANCE's *Le Monde Informatique, Distributique, InfoPC, PC Hebdo, LM Telecom;* GREECE's *Micro and Computer Age;* HUNGARY's *Computerworld/SZT, Mikrovilag, Tudomany;* INDIA's *Dataquest;* ISRAEL's *People & Computers Weekly, People & Computers Monthly;* ITALY's *Computerworld Italia;* JAPAN's *Computerworld Japan;* MEXICO's *Computerworld Mexico, Compumundo;* THE NETHERLANDS' *Computerworld Netherlands, PC World Netherlands;* NEW ZEALAND's *Computerworld New Zealand;* NORWAY's *Computerworld Norge, PC World Norge;* PEOPLE'S REPUBLIC OF CHINA's *China Computerworld, China Computerworld Monthly;* SAUDI ARABIA's *Arabian Computer News;* SOUTH KOREA's *Electronic Times/Computerworld, PC World;* SPAIN's *Computerworld Espana, Commodore World, PC World Espana, Communicaciones, Informatica Industria;* SWEDEN's *ComputerSweden, Mikrodatorn, PC World;* SWITZERLAND's *Computerworld Schweiz;* UNITED KINGDOM's *Computer News, DEC Today, ICL Today, PC Business World, Lotus UK;* UNITED STATES' *AmigaWorld, Boston Computer News, CD-ROM Review, Computer × Software News (Lebhar-Friedman), Computer Currents, Computers in Science, Computerworld, Digital News, Federal Computer Week, 80 Micro, FOCUS Publications, InCider, InfoWorld, MacWorld, Network World, PC World, Portable Computer Review, Portable 100, Publish!, PC Resource, RUN;* VENEZUELA's *Computerworld Venezuela;* WEST GERMANY's *Computerwoche, PC Welt, RUN, Information Management, PC Woche.*

```
    :RCUR=1
    :CCUR=1
    :RETURN
950 PRINT @240, "F)ill C)lear S)ave B)lank E)dit"
    :PRINT @280, "L)oad D)own T)est K)lone Q)uit M)ove";
    :RETURN 'COMM AND LINE
1000 REM DRAW A CHARACTER--ENTRY/CHAR
1005 SW=1
    :FOR RCUR=1 TO 8
    :MSK=2^(8-RCUR)
    :FOR CCUR=1 TO 11
1010 IF CD(CHAR, CCUR) AND MSK THEN GOSUB 500
1015 NEXT CCUR, RCUR
    :RETURN
1050 PRINT". . hit [enter] to BYPASS"
    :RETURN
1100 REM SAVE ROUTINE
1105 NE=1
    :F$=""
    :CLS
    :PRINT"  * *SAVE A SET OF CHARACTERS* *"
    :PRINT @85,;
    :GOSUB 1050
    :PRINT @125,;
    :INPUT "SAVE FILE";F$
    :IF F$="" THEN 1199
1108 OPEN F$ FOR OUTPUT AS 1 ELSE RETURN
1109 PRINT @165, ". . FILE BEING WRITTEN. ."
1110 FOR CHAR=0 TO 94
    :FOR CCUR=0 TO 11
    :PRINT #1, CD(CHAR, CCUR);" ";
    :NEXT CCUR, CHAR
```

of 12 columns (numbered zero to 11) by eight rows. The twelfth column is always blank. The user can specify the start and end positions that the printer will use when in proportional mode (in standard mode the full width is always printed).

Once you get the character visually the way you like it, center it appropriately in its work area so that the character will be centered when printing in non-proportional mode. Hit *E* to get back to the top of the editing sequence and re-enter the ASCII value.

Determine the first column that will be considered to start the character (it may include one or two blank columns in front of the actual character) and enter this value. Then enter the ending column (which must be at least five columns greater than the start position). The character is now ready for proportional printing. The editor will still accept data for all columns, and this data will be saved in files, but data beyond the last column specified will be ignored by the printer when in proportional mode.

Remember that both descender and length data can be changed by "re-editing" the character.

In order to keep the program logic wide open for user modification, extensive comments and mnemonic (rather than two-character) variable names were used. Obviously, the program is much larger than it needs to be. Packing it by hand or by using a packer program will cut the size down significantly if it's a problem.

The structure is strongly modular in nature, with each module defined by its comments. BASIC, by its nature, does not enforce structure (as Pascal, C or others do), but taking the effort to construct the program with a structured philosophy from the beginning helps significantly as the program expands beyond its original concept.

To keep the file size compact, I have restricted the changeable ASCII s to 33-127 — which is the range normally used by text type programs. The Epson, however, will accept 255 different characters. By changing dimensions of CD and altering the value subtracted from CHAR in lines 910 and 911, you can modify to define as many characters as desired.

A significant amount of time goes into the design of a typeface, so it's important to be able to recover in run-

```
1115 CLOSE
   :FL=0
1199 CLS
   :RETURN
1200 REM LOAD FILE
1205 NE=1
   :CLS
   :PRINT @5, "* *LOAD AN EXISTING FILE* *"
   :IF FL THEN PRINT @40, "NOTE: Current Data has been modified"
   :PRINT " since last SAVE"
1206 F$=""
   :PRINT @125, ;
   :GOSUB 1050
   :PRINT @165, ;
   :INPUT "FILE TO LOAD";F$
   :IF F$<>"" THEN OPEN F$ FOR INPUT AS 1 ELSE 1299
1208 PRINT @205, ". .FILE BEING LOADED. ."
1210 FOR CHAR=0 TO 94
   :FOR CCUR=0 TO 11
   :INPUT #1, CD(CHAR, CCUR)
   :NEXT CCUR, CHAR
   :CLOSE #1
1215 FL=0
1299 CLS
   :RETURN
1300 REM DOWNLOAD TO PRINTER
1305 NE=1
   :CLS
   :PRINT"* * DOWNLOAD TO PRINTER * *"
1307 IF(INP(187) AND 6)<>2 THEN BEEP
   :PRINT" Printer not ready . . Hit any key";
   :A$=INPUT$(1)
```

```
    :GOTO 1349
1310 GOSUB 1050
    :A1=0
    :A2=0
    :PRINT"ASCII RANGE:";
    :INPUT "from ";A1
    :INPUT " to ";A2
    :IF A1=0 OR A2=0 THEN 1349
1315 INPUT "LOAD STD CHAR SET FIRST";R$
    :IF INSTR(1, "Yy", R$) THEN LPRINT ES$;":";Z0$;Z0$;Z0$;
1318 PRINT "DOWNLOAD IN PROGRESS..."
1320 LPRINT ES$;"%";CHR$(1);Z0$;
    :LPRINT ES$;"&";Z0$;CHR$(A1);CHR$(A2);
1325 A1=A1-33
    :A2=A2-33
    :FOR CHAR=A1 TO A2
    :FOR CCUR=0 TO 11
    :LPRINT CHR$(CD(CHAR, CCUR));
    :NEXT CCUR, CHAR
1330 CLS
    :PRINT"DOWNLOAD COMPLETE"
    :BEEP
1332 PRINT" TEST (Y/N)?";
    :IF INPUT$(1)="Y" THEN GOSUB 1400
1349 CLS
    :RETURN
1400 REM TEST CHARACTERS
1405 NE=1
    :CLS
    :TS$=""
    :PRINT "* *TEST CHARACTERS* *"
    :IF(INP(187) AND 6)<>2 THEN BEEP
```

Listing continues on page 50

ning condition from an I/O error. That is why file errors are fairly extensively trapped with error messages. The more common file errors will not stop the program.

Lines 1615 through 1620 cover disk access errors. Although it's set up for Acroatix's Powr-Disk, it can be easily modified for Traveling Software's TS-DOS. Of course, it can be deleted entirely if you're not using the portable disk drive.

Line 1690 is a catch-all for miscellaneous errors. Program execution stops. If you can fix the problem by changing the value of an offending variable (*not* by changing a program line) and resuming at the same statement, the program can then be continued without losing the current work.

Note that line 1608 RESUMEs to line 100 rather than to the subroutine level of the error statement. From what I can gather, when a memory overflow error occurs, the subroutine stack is lost, causing the program to resume at its lowest level. (I've only tried this technique on the Model 100.) All other types of errors that I have encountered can be resumed on the same level from which the error occurred.□

GRAPHIC BY ELIZABETH MARITATO

Taking Command of Your Printer

Cy Callaghan

I wrote the utility program Printer Command, or PRINTC.BA, to make my Tandy DMP-105 printer more versatile. I wanted to avoid having to enter those tedious BASIC line commands to initialize its various print features. So I packaged them up, assigned function keys to each command and made the whole thing accessible from a menu.

The program was written on the Tandy 200 but can be modified to work with the Model 100/102. Similarly, it's not limited to just the DMP-105 but can be changed to accomodate virtually any printer.

I've found the program to be extremely useful for a variety of printing jobs, especially in facilitation spreadsheet work.

Since the program as written is for the Tandy DMP-105 it will operate most modern "DMP-series" dot-matrix printers, as well as all Epson dot-matrix printers, without modification. With other printers you will have to compare each command in PRINTC.BA to the same command listed in your printer's manual. Locating your printer manual will be the most difficult part of modifying this program.

The program is designed to be easy to use. It requires just pressing function keys to access commands. Lines 5 to 55 set up the screen to list the functions and indicate which key to press. Line 60 uses BASIC's KEY command to disable the keyboard and open the function keys as GOSUB commands.

PRINTER COMMAND

```
1Ø 'PRINTC.BA by Cy Callaghan
5 CLS
1Ø LINE(2Ø,5)-(219,17),1,B
15 PRINT @ 52,"PRINTER COMMAND"
2Ø PRINT @ 12Ø," F1 - Normal
   F5 - Bold Print"
3Ø PRINT @ 16Ø," F2 - Compressed
   F6 - Underline"
4Ø PRINT @ 2ØØ," F3 - Condensed
   F7 - End Elng/Bold/UL"
5Ø PRINT @ 24Ø," F4 - Elongation
   F8 - Main Menu"
55 PRINT @ 291,"(SELECT F1 - F8)"
6Ø KEY ON
8Ø ON KEY GOSUB 11ØØ, 12ØØ, 13ØØ,
   14ØØ, 15ØØ, 16ØØ, 17ØØ, 18ØØ
1ØØØ GOTO 8Ø
11ØØ LPRINT CHR$(27) CHR$(19)
     :GOTO 8Ø
12ØØ LPRINT CHR$(27) CHR$(23)
     :GOTO 8Ø
13ØØ LPRINT CHR$(27) CHR$(2Ø)
     :GOTO 8Ø
14ØØ LPRINT CHR$(27) CHR$(32)
     CHR$(27) CHR$(14):GOTO 8Ø
15ØØ LPRINT CHR$(27) CHR$(15)
     CHR$(27) CHR$(31):GOTO 8Ø
16ØØ LPRINT CHR$(15):GOTO 8Ø
17ØØ LPRINT CHR$(27) CHR$(15)
     CHR$(27) CHR$(32) CHR$(14)
     :GOTO 8Ø
18ØØ MENU
```

The commands to the printer are in lines 1100 to 1800. They correspond with function keys one through eight. The first three commands change the character font style. The three choices are *normal, compressed* and *condensed.* These changes can increase the number of characters on each line of type. *Normal* supplies 80 characters per line, *compressed* results in 96 characters and *condensed* gives you 132. To use other terms, *normal* is 10 characters per inch (cpi), or Pica; *compressed* is 12 cpi or Elite; and *condensed* is 17 cpi. You should be aware that not all printers support all of these type styles.

As you can see, the first seven command lines all end with the statement *:GOTO 80.* This returns control to the function keys for the next command. The screen display does not change when a function is executed. Most printers will indicate the command was received because it will cause a paper feed. These first three commands replace one another in the printer.

The commands on F4 and F5 are *elongation* and *bold.* They effect the width and density of the letters. *Elongation* makes the letters twice their width and *bold* causes the printer to strike a second dot next to the first, creating a more dense print.

Lines 1400 and 1500 are more complex than the previous lines. These

commands will not be accepted if the other one is already in the printer. Therefore, the first code on F4 (line 1400) is one that cancels *bold* so *elongation* can be put in. If you have to revise the program for your printer, keep this aspect in mind.

PRINTC.BA's versatility comes from these commands. Since you can select three different letter sizes and three different type faces, you can combine them to give nine different looks to the print! For instance, *condensed elongation.*

F6 (line 1600) turns on the underline function. This takes a little practice to have any good results. Usually I insert a separate text file into the letter for the underlined portion.

F7 is a clearing function that removes the commands *elongation, bold* and *underline.* This is handy if you have inserted a paragraph using one of these functions and then want to return to the previous form to continue the letter. Your printer will probably default to normal print when you turn it off and then turn it on again.

As I mentioned earlier you'll have to change a couple of the lines in the listing for running this program on a Model 100. Those lines are:

```
5 CLS:CALL 16959
1800 CALL 16964:MENU
```

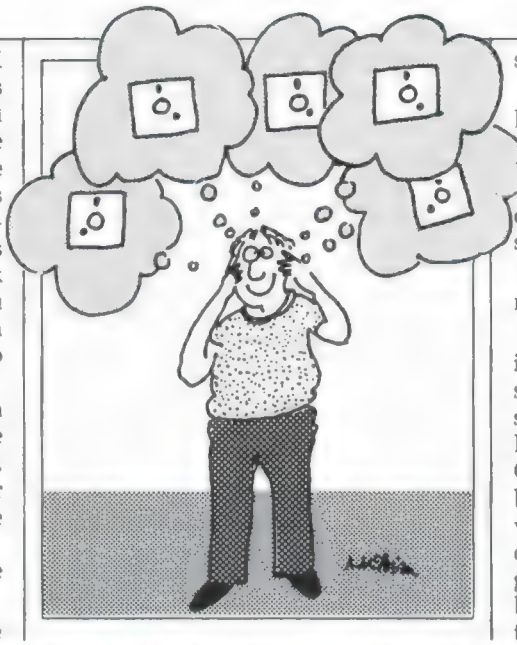
This prevents the screen's line advance from moving PRINTC.BA's display.

Let's talk letter-quality printing. The only way I can get what appears to be letter quality print from a DMP-105 is by having six beers before I call for a printout. But I have found that the bold print with normal font style results in a letter I find satisfactory for everyday correspondence.

Using PRINTC.BA with spreadsheets is probably the most rewarding aspect of the program for me. I use a Tandy 200 with Multiplan. Using Multiplan's normal column width of nine characters, eight columns can be printed at 80 characters across. With PRINTC.BA, however, I can set the type font style to condensed. Multiplan will then print at a width of 155 characters, or 14 columns. This makes a tremendous difference in the usefulness of Multiplan as a tracking and analysis tool. I assume it would have equal results with any spreadsheet program.

In addition to all its features, one of the best thing about PRINTC.BA is that it is only 544 bytes long. That means you can leave it in RAM without sacrificing too much memory.□

Crazy Like a Box



Some time ago when my wife and I needed something to keep the kids occupied — while waiting for our food to come at the restaurant, for instance — we made good use of a very simple and commom game. All that was needed was a scrap of paper and a pencil. An orderly array of a few dozen dots was quickly drawn, and the kids took turns connecting pairs of dots. If you closed a square, you put your initials in it and kept your turn. The object was to close as many squares as possible.

I don't know if the game ever had a name, but it was an effective pastime for children forced to wait and be quiet. Recently, since this is the computer age, I decided to see what it would take to teach the game to my Model 100.

The basic program has roughly five different sections:

1. An instruction panel, put on the screen by lines 30 to 95.

2. The playing board produced in lines 100 to 140.

3. The input section, lines 210 to 290.

4. The section in which the dots are connected, the boxes closed and the score maintained (lines 1000 to 4220).

5. The posting of the score and error messages (lines 4230 to 10020).

The call to 16959 in line 5 uses a built-in ROM routine to freeze the vertical scrolling of the display. To restore scrolling after a game, exit BASIC to Edit or Menu or call ROM 16964. The CLEAR 1000 in line 5 sets aside 1,000 bytes for string variables. Two string variables, B$ and P$, are utilized and continuously changed throughout the game. B$ holds the coordinates of the box selected by a player. P$ identifies the player as either No. 1 or No. 2. Ap-

GAME.BA: A game that may make you crazy as a box.

```
1 'GAME.BA 6/27/86 E. Carmody
5 CALL 16959
 :CLEAR 1000
10 DEFINT C, F, I, N, P, S, X, Y
  :DIM A(40, 5), S(2)
  :P=1
30 CLS
  :CALL 17001
  :PRINT @0, "cRaZy LikE a BOX";
  :CALL 17006
40 PRINT @80, "Two players take turns.    ";
   CHR$(227); " 1 "; CHR$(228)
50 PRINT @120, "Connect dot-pairs to draw  4   ";
   CHR$(234)
60 PRINT @160, "sides and close boxes.       ";
   CHR$(234); " 2"
70 PRINT @200, "Select boxes, sides by     ";
   CHR$(225); " 3 "; CHR$(226)
80 PRINT @240, "coordinates - e.g. a3, 2    BOX a3"
90 PRINT @280, "Close box & keep turn.    SIDE 2"
95 FOR I=1 TO 7500
  :NEXT I
100 CLS
   :CALL 17001
   :PRINT @0, "cRaZy LikE a BOX";
   :CALL 17006
   :PRINT @25, "abcdefgh"
110 FOR X=1 TO 5
   :PRINT @22+(X+1)*40, X
   :NEXT X
120 FOR Y=15 TO 55 STEP 8
   :FOR X=149 TO 197 STEP 6
130 PSET (X, Y)
   :PSET (X+1, Y)
   :PSET (X, Y+1)
   :PSET (X+1, Y+1)
140 NEXT X
   :NEXT Y
210 S=0
   :F1=0
   :F2=0
   :PRINT @80, "PLAYER"; P
   :BEEP
   :INPUT "BOX, SIDE"; B$, S
220 C1=ASC(LEFT$(B$, 1))
230 C2=VAL(RIGHT$(B$, 1))
240 IF C1<97 OR C1>104 OR C2<1 OR C2>5 OR S<1 OR
    S>4 THEN 10000
250 N=(C1-72)+(C2+1)*40  'CURSOR POS
260 N1=N-(72+C2*32)     'BOX#
270 IF A(N1, S)=1 OR A(N1, 5)=4 THEN 10000
280 X=(NMOD40)*6
290 Y=8*INT(N/40)
300 ON S GOTO 1000, 2000, 3000, 4000
1000 LINE(X, Y-1)-(X+5, Y-1)
    :LINE(X, Y)-(X+5, Y)
    :P$=RIGHT$(STR$(P), 1)
1010 A(N1, 1)=1
    :A(N1, 5)=A(N1, 5)+1
1020 IF C2=1 THEN 1040
1030 A(N1-8, 3)=1
    :A(N1-8, 5)=A(N1-8, 5)+1
1040 IF A(N1, 5)=4 THEN BEEP
    :PRINT @N, P$
```

parently, the standard 256 bytes normally reserved when entering BASIC was not enough, and the computer locked up after 48 turns. The calls to 17001 and 17006 in line 30 turn on and off the reverse printing mode to emphasize the title.

The board consists of 40 boxes, identified by a letter-number coordinate system — *a* through *h* across the top, *1* through *5* vertically. Note that lower case letters are used. Each box in the array consists of a full character position of six-by-eight pixels. Four pixels, one each from the corners of adjoining boxes, form the dots. This is done in line 130.

Line 210 requests and accepts player input. After each player's turn it returns to this line for the next move. Sides (the variable S) are labeled 1 through 4, clockwise from the top. A comma must separate the box and side input.

Lines 220 to 290 translate the input to LCD *x* and *y* coordinates and check for input error. C1 is the ASCII value of the letter coordinate of the selected box. C2 is the number coordinate. N is the LCD cursor position of the box, and is converted to N1, a box number of 1 through 40.

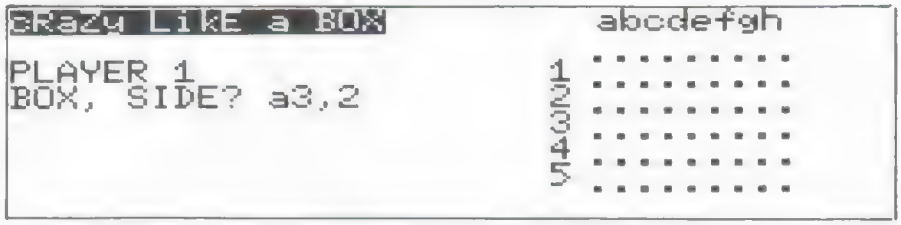
The formulas in 280 and 290 find the upper left pixel x and y values for N. An array A(40,5) keeps track of which and how many sides of each box are selected, and is confirmed by line 270. If a previously selected side is chosen in error, the program jumps to the error routine in line 10000.

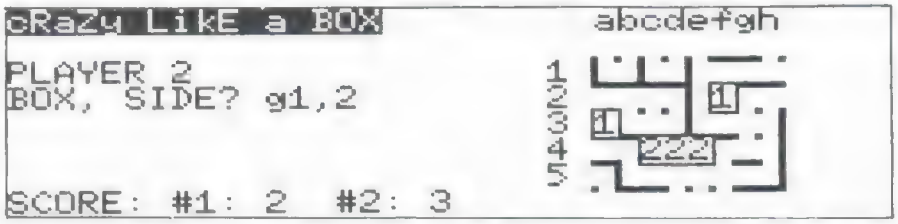
Based on which side was selected (1, 2, 3 or 4), execution branches to one of four similar procedures. We will follow a selection of side one (1000 to 1230). First, the chosen line is drawn and the array is updated. Unless the selected box is at the edge of the board, the array is also updated for the appropriate adjacent box (1000 to 1030).

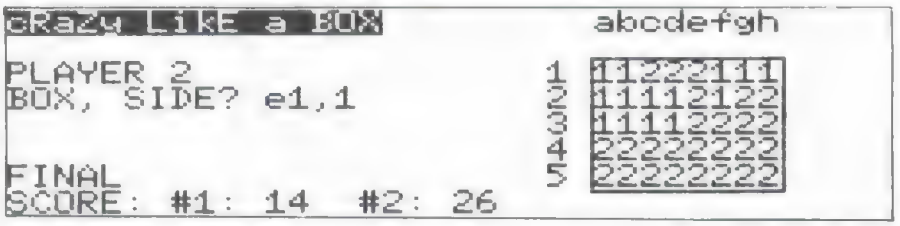
For example, if side one of box c3 is chosen, the array is also updated for box c2, side three. Next, the array column 5, which keeps track of the number of sides drawn for each box, is checked for the value 4. This also is done both for the box selected and the adjacent box, if there is one. If *4* is found, the player number P$ is printed within the box and the score is updated in lines 1040 to 1050. (A string variable and the formula for it listed in line 1000 are used so a number can be printed

```
    :S(P)=S(P)+1
    :F1=1
1045 IF C2=1 THEN 1060
1050 IF A(N1-8, 5)=4 THEN BEEP
    :PRINT @N-40, P$
    :S(P)=S(P)+1
    :F2=1
1060 IF F1=1 OR F2=1 THEN 1210
1070 IF P=1 THEN P=2
    :PRINT @120, SPACE$(20)
    :GOTO 210
1080 IF P=2 THEN P=1
    :PRINT @120, SPACE$(20)
    :GOTO 210
1210 FOR I=1 TO 40
    :IF A(I, 5)<>4 THEN PRINT @120, SPACE$(20)
    :GOSUB 5000
    :GOTO 210
1220 NEXT I
1230 GOTO 4230
2000 LINE (X+5, Y)-(X+5, Y+7)
    :LINE (X+6, Y)-(X+6, Y+7)
    :P$=RIGHT$(STR$(P), 1)
2010 A(N1, 2)=1
    :A(N1, 5)=A(N1, 5)+1
2020 IF C1=104 THEN 2040
2030 A(N1+1, 4)=1
    :A(N1+1, 5)=A(N1+1, 5)+1
2040 IF A(N1, 5)=4 THEN BEEP
    :PRINT @N, P$
    :S(P)=S(P)+1
    :F1=1

2045 IF C1=104 THEN 2060
2050 IF A(N1+1, 5)=4 THEN BEEP
    :PRINT @N+1, P$
    :S(P)=S(P)+1
    :F2=1
2060 IF F1=1 OR F2=1 THEN 2210
2070 IF P=1 THEN P=2
    :PRINT @120, SPACE$(20)
    :GOTO 210
2080 IF P=2 THEN P=1
    :PRINT @120, SPACE$(20)
    :GOTO 210
2210 FOR I=1 TO 40
    :IF A(I, 5)<>4 THEN PRINT @120, SPACE$(20)
    :GOSUB 5000
    :GOTO 210

2220 NEXT I
2230 GOTO 4230
3000 LINE (X, Y+7)-(X+5, Y+7)
    :LINE (X, Y+8)-(X+5, Y+8)
    :P$=RIGHT$(STR$(P), 1)
3010 A(N1, 3)=1
    :A(N1, 5)=A(N1, 5)+1
3020 IF C2=5 THEN 3040
3030 A(N1+8, 1)=1
    :A(N1+8, 5)=A(N1+8, 5)+1
3040 IF A(N1, 5)=4 THEN BEEP
    :PRINT @N, P$
    :S(P)=S(P)+1
    :F1=1
3045 IF C2=5 THEN 3060
3050 IF A(N1+8, 5)=4 THEN BEEP
```

```
    :PRINT @N+40, P$
    :S(P)=S(P)+1
    :F2=1
3060 IF F1=1 OR F2=1 THEN 3210
3070 IF P=1 THEN P=2
    :PRINT @120, SPACE$(20)
    :GOTO 210
3080 IF P=2 THEN P=1
    :PRINT @120, SPACE$(20)
    :GOTO 210
3210 FOR I=1 TO 40
    :IF A(I, 5)<>4 THEN PRINT @120, SPACE$(20)
    :GOSUB 5000
    :GOTO 210
3220 NEXT I
3230 GOTO 4230
4000 LINE (X-1, Y)-(X-1, Y+7)
    :LINE (X, Y)-(X, Y+7)
    :P$=RIGHT$(STR$(P), 1)
4010 A(N1, 4)=1
    :A(N1, 5)=A(N1, 5)+1
4020 IF C1=97 THEN 4040
4030 A(N1-1, 2)=1
    :A(N1-1, 5)=A(N1-1, 5)+1
4040 IF A(N1, 5)=4 THEN BEEP
    :PRINT @N, P$
    :S(P)=S(P)+1
    :F1=1
4045 IF C1=97 THEN 4060
4050 IF A(N1-1, 5)=4 THEN BEEP
    :PRINT @N-1, P$
    :S(P)=S(P)+1
    :F2=1
4060 IF F1=1 OR F2=1 THEN 4210
4070 IF P=1 THEN P=2
    :PRINT @120, SPACE$(20)
    :GOTO 210
4080 IF P=2 THEN P=1
    :PRINT @120, SPACE$(20)
    :GOTO 210
4210 FOR I=1 TO 40
    :IF A(I, 5)<>4 THEN PRINT @120, SPACE$(20)
    :GOSUB 5000
    :GOTO 210
4220 NEXT I
4230 GOSUB 5000
    :BEEP
    :BEEP
    :PRINT @240, "FINAL";
    :GOTO 4230
    :END
5000 PRINT @280, "SCORE: #1:"; S(1); " #2:"; S(2)
5010 RETURN
10000 BEEP
    :PRINT @120, "INPUT ERROR      "
    :BEEP
    :BEEP
    :FOR I=1 TO 500
    :NEXT I
    :BEEP
10010 IF P=1 THEN P=2
    :GOTO 210
10020 IF P=2 THEN P=1
    :GOTO 210
```







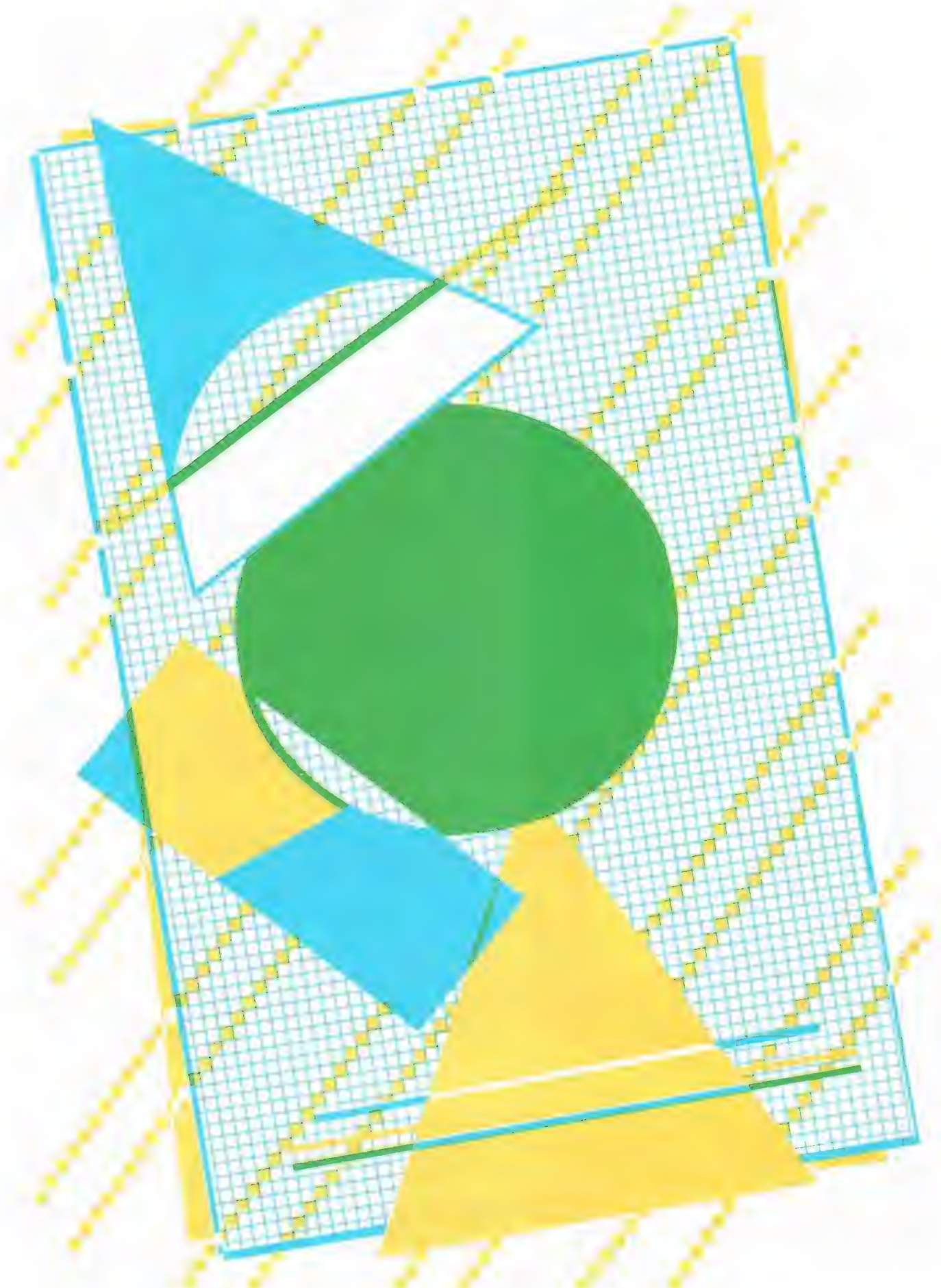
without the leading space that would otherwise occur.)

Now all the boxes in the array are checked to see if the game is over (lines 1210 to 1230). If a box is not closed by a player, the player number is alternated and operation is returned to line 210 for another input. If a box has been closed but the game is not over, the player number is not reversed.

Line 5000 prints the score at the bottom of the screen every time one or two boxes are closed. The error routine (10000 to 10020) is used to punish the offender as well as reject inputs. A player loses his turn if he commits an error by indicating upper case instead of lower case letters, picking wrong coordinates or selecting a side previously taken.

Working out the program was an interesting challenge and caused a bit of head scratching for a while — especially when I could not figure out why the computer locked up after 48 turns and everything worked fine up until then. Maybe somebody else wants to try Version 2, in which one of the opponents is the computer itself. Crazy!

—Emmett J. Carmody



The Incredible Portable Sketchpad

A World of Graphics for your Portable.

By Mark Laster

Many of you are familiar with the standard way of creating bit-mapped graphics on the Model 100 display — laborious trigonometric calculations for drawing circles, logarithmic lines and plentiful PSETs and PRESETs. Well, just as a word processor can save you from manually formatting paragraphs, so can a general-purpose graphics package help you cope with pixels. And that's the purpose of SHAPES.BA.

With Shapes, you can create a graphics pattern on the screen using several graphics primitives (point, line, triangle, rectangle, circle). The pictures can then be saved to a document (.DO) RAM file for further use, or output to a printer.

Furthermore, you don't have to work with *x-y* coordinates. Simply move the one-pixel cursor around the LCD, selecting points and operations as you go. Voila! A work of art.

ENOUGH HYPERBOLE

With Shapes, which occupies about 8700 bytes, you can manipulate a cursor from the keyboard, and can define up to four locations as coordinate pairs. Once the coordinates are defined, three of them can be the vertices of a triangle; two of them can be either the ends of a line segment (which also can be the radius of a circle) or the diagonally opposite ends of a rectangle, and one of them can be an individual pixel. The user can blacken or whiten a circle, rectangle, triangle, line or pixel.

By the way, the word *pixel* is a contraction of *picture element*. One of the little square LCD dots on your computer's screen is a single pixel; in text mode, a six-by-eight block of pixels is used to display a character. Another word for block is *matrix*, and another word for pixel is *dot*, hence the term *dot-matrix*.

Shapes runs in the Model 100's BASIC, without peeks or pokes. Within the program, PSET, PRESET and LINE commands translate the user's keystrokes into effects on the screen. That makes the Shapes fully Tandy 102-compatible and easily translated to run on the Tandy 200.

Shapes divides the screen into the graphic area on the left, 75 pixels wide by 64 high, and the textual area on the right. The graphic area of the screen is bit-mapped in an integer array named CO (not to be confused with the .CO machine-language file extension, by the way). The program interrogates or assigns values to the two-byte words of the array via word- or bit-oriented operations. A bit value of binary zero corresponds to a white pixel; a binary one is a black pixel.

GETTING PLACES

The cursor (or the pointer to the current graphic element) is a single pixel. There are two sets of keys for cursor control located on the left and right sides of the alphabetic keyboard. Those on the right, meaning the eight keys roughly adjacent to the letter **L**, move the cursor 15 columns or rows if the key is shifted, and seven if unshifted. The **L** itself produces no movement in any direction; by pressing it, the user can blacken the current cursor location.

Characters on the left side of the keyboard move the cursor only one column or row at a time. Unshifted characters cause pixels to be blackened until the user resets them, but shifted characters on the left blacken and then automatically whiten pixels. As a result, the shifted keys on the left can work as an eraser, resetting pixels (and zeroing the associated bits in array CO); in particular, the shifted **S** erases whatever location the cursor occupies. Character strings in program lines 5, 6 and 7 identify the relevant keys and the directions of cursor movement. The string A1 contains the keyboard characters; A2, the corresponding vertical movement; A3, the horizontal movement. The "-" means upward in A2, and in A3, it means to the left. Zeros are placeholders, and "+" works opposite of "-." The variable B4 carries a "+" value if the shape or pixel is to be black, and "-" if white.

PAIRING OFF

There are four points, or *x*- and *y*-coordinate pairs that you can use to specify special operations. (Remember the old expression that two points determine a line, three a triangle?) After you move the cursor to a screen location that you wish to store as a coordinate pair (say, one end of a line segment), press one of the number keys in the top row: **1**, **2**, **3** or **4**; if you do, the current cursor position will replace whichever pair of coordinates had been associated with the same number key **1** through **4**. The right side of the screen displays numeric coordinates. Those locations that appear at the upper left part of the screen are displayed highest on the list.

Another way to move the cursor is to define its position (if the cursor is already at a defined coordinate set) as one of the other established coordinate sets. To do this, press **@**. In response to a prompt, type a key in the range **1** through **4** as the new position for the cursor.

Drawing a black line segment is a three-keystroke operation. First, press **5**; then, in response to a prompt, press the key that refers to the starting coordinate pair (one of the keys: **1** through **4**), and then press the key that refers to the ending coordinate pair (any of **1** through **4** except the preced-

ILLUSTRATION BY JOE RYAN

ing keystroke). Drawing a white line segment involves the same steps, except for starting with % instead of 5.

Table 2 contains a summary of commands for producing this shape, as well as the triangle, rectangle, circles and lines produced by quick-draw.

Within Shapes, the line-drawing statements begin at line 50. If a line will cover more horizontal distance than vertical, a loop-index that corresponds to shallow slope becomes active, stepping one column of pixels with each pass through the loop; however, if a line will cover more vertical distance than horizonal, the steep index becomes the effective one. This alternate-index scheme draws line segments that are smoother for the complete range of slopes than a single-index scheme can create. If a line is horizontal only, lines that start at label 600 deal with it.

TRIANGLES AND CIRCLES

Calculation for triangles may take up to 30 seconds before the pattern emerges. Pixels in the second vertical bar, column 76, indicate the current line of activity, after the pattern starts to appear.

The smallest area that the rectangle-drawing function permits is a single pixel; the next smallest is a vertical or horizontal line segment, one pixel thick. By defining diagonally opposite corners of the graphic area as corners of the rectangle, it is possible to blacken or whiten that 75-by-64 area in one operation.

The BASIC code for this operation starts at line 70, calling subroutines that use the LINE command. To create a solid shape, LINE creates adjacent horizontal rows; the endpoints of each row comprising the shape are stored in arrays LL and RR. Values are set into those arrays at line 710, and are accessed at line 810.

Shapes draws four types of circle: solid black, solid white, hollow black and hollow white. If points of a circle fall outside the graphic area, the program doesn't crash; those points are simply ignored. The periphery of the circle consists of a variable number of points, depending on the length of the radius. This sliding scale keeps large circles free of gaps, and keeps small circles from requiring as much preparation time as large circles.

Table 1. Commands for Cursor Manipulation and Coordinate Definition

FUNCTION	FIRST KEY
Move Cursor One Column and/or Row, and Leave Pixel Black	Any one of these: *a, q, w, e, d, c, x, z;* (unshifted).
Move Cursor One Column and/or Row, and Leave Pixel White	Any one of these: *A, Q, W, E, D, C, X, Z;* (shifted).
Move Cursor Seven Columns and/or Rows, and Leave Pixel Black	Any one of these: *k, i, o, p, ;, ', ., ,;* (unshifted).
Move Cursor Fifteen Columns and/or Rows, and Leave Pixel Black	Any one of these: *K, I, O, P, :, ?, >, <* (shifted).
Whiten Present Cursor Location	Shifted *S*
Blacken present Cursor Location	Shifted *L*, or Unshifted *l* or *s*
Define a Coordinate Pair	Any one of these: *1,2,3,4*
Move Cursor to a Defined Coordinate Pair	' Then, press any one of: *1, 2, 3, 4*

ONE MORE

Unlike the other functions described above, the quick-draw operation does not require a fixed number of keystrokes. To start it, type the exclamation point ! ; then in response to a prompt, furnish the coordinate-pair reference, that is, one of the keys: **1** through **4**. Whichever coordinate pair you referenced becomes one end of a line segment; let's call it the anchored end. The other end (the free end) responds to one or more consecutive keystrokes on either side of the keyboard, as previously described. The program draws, and then erases the line segment in response to a keystroke.

With each keystroke the line is drawn and erased. To replace a vanished but correctly positioned line with a permanent one, press a number key (**1** through **4**) other than the anchor point; then perform the standard procedure to draw a line between two defined coordinate sets, starting with pressing the **5**.

Another distinction between the quick-draw operation and the others,

Table 2. Shape-Construction Commands

SHAPE	COLOR	FIRST KEY	REMAINDER OF COMMAND
Line	Black	5	Press any two different keys of the set: *1, 2, 3, or 4* representing the coordinate pairs of the ends of the line segment.
	White	%	
Tri-angle	Black	6	Press any three different keys of the set: *1, 2, 3, or 4* representing the coordinate pairs of the vertices of the triangle.
	White	^	
Rec-tangle	Black	7	Press any two different keys of the set: *1, 2, 3, or 4* representing the coordinate pairs of the diagonally opposite corners of rectangle.
	White	&	
Hollow Circle	Black	0	Press any two different keys of the set: *1, 2, 3, or 4* representing the coordinate pairs of the center, and a point on the rim.
	White	)	
Quick-Draw Line	Black, then White	!	Press any of the coordinate keys you can press the same key or a different key, any number of times. To end, either redefine a coordinate set, or start another command.

Powerful New Programs for your Radio Shack Model 100/102/200

DO NOT ADJUST YOUR SCREEN...

... nothing is wrong with your Model 100/102 that the new ULTRASCREEN™ program won't cure.

Normally, the Radio Shack™ Model 100/102 offers a screen with only a 320-character capacity - 40 characters per line, by 8 lines. Furthermore, if a word is too big to fit at the end of a line, the built-in Text editor shifts the ENTIRE WORD to the next line (this is called "word wrapping"). While this makes the text easier to read (no split words), too much screen space is WASTED at the right of almost every line.

Now, ULTRASCREEN gives you the power of a 60-character by 10-line screen, thereby correcting the main design flaw of your Model 100/102 - the small screen. With no decrease in the legibility of the characters, ULTRASCREEN more than doubles the effective viewing capacity of the 100/102: it increases the total number of characters in the display from 320 to 600, and reduces the number of word wraps by up to 1/3!

```
Additional features of ULTRASCREEN 100/1
02:◀
- Written entirely in machine language,
for speed of operation and minimum RAM
usage (less than 2.6K).◀
- Relocatable in memory - no problem to
use in conjunction with other machine
language programs.◀
```

```
Additional features of ULTRASCREEN 100/102:◀
- Written entirely in machine language, for speed of
operation and minimum RAM usage (less than 2.6K).◀
- Relocatable in memory - no problem to use in conjunction
with other machine language programs.◀
- Full Model 100/102 character set is supported, including
GRAPHICS and INTERNATIONAL CHARACTERS.◀
- Can be used from all built-in programs (eg. BASIC, TEXT,
TELCOM, etc.), as well as your own BASIC or M/L programs.◀
- ESPECIALLY SUITED FOR USE WITH TEXT POWER 100/102.◀
```

FOR CATALOGUE, WRITE:

ULTRASOFT INNOVATIONS, INC.
76 Main Street / P.O. Box 247
Champlain, NY 12919-0247
ORDER LINES:
(800) 448-4511 (U.S. except N.Y.)
(800) 962-1480 (N.Y. state)
(800) 255-1133 (Ont. and Que.)
Ask for Extension 313-18
For product information, call:
(514) 457-9293

Orders shipped within 7 days of receipt (you will be informed of any undue delay). Items ordered together are not necessarily shipped the same day. TEXT POWER, PAGE PLOT and MERGE TEXT are registered trademarks of Hugo Ferreyra. Copyright ©1987 by UltraSoft Innovations, Inc.

YOU DIDN'T GET WHAT YOU THOUGHT YOU WERE GETTING, DID YOU?

If you bought a Model 100/102 for word processing, you were looking for CONVENIENCE, FLEXIBILITY and PORTABILITY. The problem was, the built-in TEXT program was only HALF a word-processor - it allowed you to enter the data into your computer and edit it, but didn't help you at all to set up that page in an attractive format for printing.

Now, TEXT POWER™, "the best formatter there is for the Model 100", according to reviews in both PORTABLE 100 and CREATIVE COMPUTER magazines, is the key which will unleash the capacity of your 100/102 to give you:

- the POWER to **create or edit text files, and save them to disk or cassette**, without ever leaving the program,
- the POWER to **format text automatically**, with tremendous speed,
- the POWER to **use all the familiar built-in TEXT features,** such as PASTE, CUT, COPY, etc.
- the POWER to **insert text at specified locations in the master text file,** using MERGE TEXT™ (e.g. insert names and addresses in a business letter),
- the POWER to **add headers and footers**, which can be turned on or off (or changed) from within the text,
- the POWER to **right-justify text,** and/or to **center text vertically or horizontally,**
- the POWER to **control left and right margins for the entire text, or for individual paragraphs,**
- the POWER to **number pages automatically in any location,** and
- the POWER to **preview the complete layout of your formatted text graphically, without embedded commands, before printing** (using PAGE PLOT™:

Each version of TEXT POWER is customized for a particular printer, or group of printers, allowing the same commands to be used, even if the actual printer code varies from one machine to another. For example, you can use the following features (individually or in combination, if your particular printer supports them): boldface, condensed, double-strike, elite type, near-letter-quality (NLQ), superscript and subscript, italics, underlining, etc.
TEXT POWER also gives you the ability to send automatic line feeds to your printer, and to advance the paper to the top of the next page, from within the text.
Optional support is available for INTERNATIONAL CHARACTER SETS - including French, Spanish, German, etc. - for use with virtually all printers.

YES! PLEASE SEND ME THE PRODUCTS INDICATED AT RIGHT.

NAME
ADDR.
CITY
STATE ZIP
TEL. (DAY)
PAYMENT BY: ☐ CHK/MO ☐ C.O.D. ☐ VISA/ ☐ MC
EXP.
SIGNED

QTY	DESCRIPTION	PRICE	TTL
	ULTRASCREEN	$34.95	
	TEXT POWER	$59.95	
	8K CHIPS-100	$22.95	
	8K CHIPS-102	$ 9.95	
MODEL: ☐ 100/ ☐ 102			
TEXT POWER ORDERS: Printer: ☐ Parallel/ ☐ Serial			
	SUBTOTAL		
	SHIPPING/HANDLING		4.00
	C.O.D. ($3.00)		
	TOTAL		

is that it does not change the bit pattern in the CO array. Therefore, if this operation draws and erases a line segment, and if in the process it erases pixels that were black prior to the quick-draw operation, the bits in CO that correspond to those pixels remain set to 1. To re-blacken the correct pixels, save the CO array to a file by using the **$** key, and load that file with the **#** key, as explained in the following paragraphs.

Table 3. Full-Screen or File-Transfer Commands

FUNCTION	FIRST OR ONLY KEY	REMAINDER OF COMMAND
Save CO Array to a .DO File	$	Filename, ending in .DO
Load .DO File, and Display on Screen	#	Filename, ending in .DO
Print CO Array	*	Choose printer: *I, T,* or *X*
Invert CO Bits and Graphic Tonality	*8*	(None)

SAVING YOUR MASTERPIECE

To save a pattern to a file, press **$**, and in response to the prompt, name the output file. Do not press the Enter key after the last letter of the .DO in the filename, or after any other entries for this program. Line 40 is the beginning of code for file output. The size of a file ranges from 1600 bytes to nearly 2900, depending on the contents. Even though Shapes manipulates bits, the output file contains spaces and integer numbers; five-digit numbers occupy more bytes than one-digit numbers.

To print a screen image in the one-to-one (1:1) aspect ratio, press the * key, and, in response to the prompt, key in **I** or **T** for IBM (Epson) or Tandy-compatible dot matrix printers, respectively; key in **X** for printers incompatible with either, or for daisy-wheel printers. Within Shapes, the relevant action starts at line 80. To print on devices that do not respond to the IBM/Epson or Tandy-style commands, you may have to adjust the device or the program or both. For example, on one particular daisy-wheel typewriter/printer, there is no half-space mode, but I found command to turn the platen half a space after printing a line.

With horizontal and vertical spacing each 12 characters/inch, the program can print a pattern whose size exceeds six inches wide by five inches high.

To invert the screen tones of the graphic area, just press **8**. Starting at line 120, the program erases the graphic area, inverts the bits of the CO array, except for high bits, and then creates a new pixel pattern based on the new values in CO.

If the pattern on the screen and the coordinate pairs seem not worth saving, you can erase the screen by leaving the program and restarting it. To do this, press Esc; this action puts you in BASIC. To restart the program from here, press F4 (not **4**, which is diagonally below it); otherwise to return to the menu screen, press F8.

If you wish to copy the graphics data file created by this Shapes to another device, such as diskette cassette, follow the usual procedure for copying any .DO file.

Look out, Da Vinci — Here comes Shapes!□

Mark Laster is a freelance technical writer from Annapolis, Maryland.

```
1 MAXFILES = 1: DEFSTR A,B: DEFINT
  C-R,T-Z: DEFSNGS: AN =
  "1234!@#$5%6^7&8*9(Ø)"
2 DIM CO(4,63), GR(4), LL(63), RR(63),
  XX(6), YY(6), SY(2): MC = 74: E$ =
  CHR$(27): CLS
3 I1 = 2: MX = 32767: PRINT @8Ø,"SHAPES":
  PRINT @16Ø,"Copyright 1986 by Mark
  Laster"
4 BK = CHR$(225)+ CHR$(247)+ CHR$(229)+
  CHR$(92): GOSUB 2ØØ: GOSUB 2ØØ
5 CLS: LINE(75,Ø)-(76,63),1,BF: A1 =
  "QqWwEeAaSsDd2zXxCcIOPKL;<>?iopkl;,./"
  ' chrs
6 AH = SPACE$(22): A2 =
  "------ØØØØØØ+++++++---ØØØ+++---ØØØ+++"
  ' vertical
7 A3 =
  "--ØØ+++--ØØ+++--ØØ++-Ø+-Ø+-Ø+-Ø+-Ø+-Ø+"
  ' horizontal
8 X = 37: Y = 32: FOR N = 1 TO 4: XX(N) =
  37: YY(N) = 32: NEXT N: XY = 1: GOSUB
  37Ø
9 A = INPUT$(1): DI = Ø
1Ø IF A = E$ THEN 999 ELSE GOSUB 37Ø
11 B4 = "+": GOSUB 31Ø: IF P = Ø THEN 15
12 IF (P>18 OR P MOD 2 = 1) THEN 13 ELSE
  B4 = "-"
13 X5 = X: X6 = X: Y5 = Y: Y6 = Y: GOSUB
  15Ø: GOSUB 37Ø: GOTO 9
15 M = INSTR(AN,A): IF M = Ø THEN BEEP:
  GOTO 9
16 IF M>Ø AND M<5 THEN XX(M) = X: YY(M) =
  Y: PSET(X,Y): GOSUB 37Ø: GOTO 9
19 '
2Ø ' <@>: switch cursor
21 IF A<>"@" THEN 3Ø ELSE PRINT
  @255,"Which point: 1,2,3,4"
22 BN = INPUT$(1): PRINT @255, SPACE$(2Ø)
23 M = INSTR(AN,BN): IF M = Ø THEN BEEP:
  GOTO 9 ELSE IF M>4 THEN 3Ø
24 X = XX(M): Y = YY(M): GOSUB 37Ø:
  GOTO 9
29 '
3Ø ' <#>: file load
31 IF A<>"#" THEN 4Ø ELSE PRINT @13,
  "Load          ": GOSUB 215: IF UF = 1
  THEN 9
32 OPEN AF FOR INPUT AS 1: FOR Y = Ø TO
  63: PRESET(76,Y)
33 FOR W = Ø TO 4: INPUT#1, CO(W,Y): NEXT
  W: NEXT Y: CLOSE
34 FOR I = 1 TO 4: XX(I) = 37: YY(I) =
  32: NEXT I: GOSUB 9ØØ: GOSUB 13Ø:
  GOTO 9
39 '
4Ø ' <$>: file save
41 IF A<>"$" THEN 5Ø ELSE PRINT @13,"Save
```

```
    to       ": GOSUB 215: IF UF = 1 THEN 9
42  PRINT @20,"Saving": PRINT
    @60,"Screen": PRINT @100,"To File":
    PRINT @180,AF
43  OPEN AF FOR OUTPUT AS 1
44  FOR Y = 0 TO 63: PRESET(76,Y)
45  FOR W = 0 TO 4: PRINT #1,CO(W,Y): NEXT W
47  NEXT Y: LINE(76,0)-(76,63),1
48  CLOSE: Y = 32: X = 37: GOSUB 900:
    GOSUB 370: GOTO 9
49  '
50  ' <5>; <%>: line BLACK, WHITE
51  Q3 = INSTR ("5%",A): IF Q3 = 0 THEN 60
    ELSE LJ = 2: PRINT @255,CHR$(222-
    Q3*41)
52  PRINT @257,"Line endpoints": A8 =
    INPUT$(2): PRINT @255,SPACE$(19)
53  GOSUB 290 ' check for 1-4, no dupes
54  IF IR = 1 THEN IR = 0: BEEP: GOTO 9
    ELSE GOSUB 770 ' build x5,x6,y5,y6
55  GOSUB 400: IF DI = 0 THEN GOSUB 190
56  GOSUB 370
57  PRINT @257,AH: GOTO 9
59  '
60  ' <6>; <^>: triangle BLACK, WHITE
61  Q3 = INSTR("6^",A): IF Q3 = 0 THEN 70
    ELSE GOSUB 750
62  GOSUB 310: PRINT @255, CHR$(251+Q3);
    " Which 3 points?    "
63  A8 = INPUT$(3): PRINT @255,
    CHR$(251+Q3); " ";A8;SPACE$(18): IR =
    0: LJ = 3: GOSUB 290
64  IF IR<>0 THEN IR = 0: BEEP: GOTO 9
65  A8 = A8 + LEFT$(A8,1): FOR NN = 1 TO
    3: N5 = VAL( MID$(A8,NN,1))
66  N6 = VAL(MID$(A8,NN+1,1)): X5 =
    XX(N5): X6 = XX(N6): Y5 = YY(N5): Y6 =
    YY(N6)
67  GOSUB 400: IF DI = 0 THEN GOSUB 190
68  NEXT NN: GOSUB 800: BEEP: GOTO 9
69  '
70  ' <7>; <&>: rectangle BLACK, WHITE
71  Q3 = INSTR("7&",A): IF Q3 = 0 THEN 80
72  PRINT @255, CHR$(223+16*Q3);
    " Which two points?": A8 = INPUT$(2):
    LJ = 2: IR = 0
73  GOSUB 290: PRINT @255,SPACE$(24): IF
    IR<>0 THEN IR = 0: GOTO 9
74  GOSUB 770: FOR I = 0 TO 63: LL(I) =
    -1: RR(I) = -1: NEXT I ' 770: build
    xx(5), etc
75  MN = X5: MP = X6: IF X6<X5 THEN MN =
    X6: MP = X5
76  MU = Y5: MV = Y6: IF Y6<Y5 THEN MU =
    Y6: MV = Y5
77  FOR Y = MU TO MV: LL(Y) = MN: RR(Y) =
    MP: NEXT Y
78  GOSUB 800: X = XX(6): Y = YY(6): GOSUB
    370: GOTO 9
79  '
80  ' <*>: file print
81  IF A<>"*" THEN 90 ELSE GOSUB 900:
    PRINT @13,"Printer ! Key In": BP = "0"
82  PRINT @53,"--------!--------": PRINT
    @95,"TANDY ! T"; PRINT
    @135,"IBM   ! I"
83  PRINT @175,"other ! X": BP =
    INPUT$(1): GOSUB 910: FOR Y = 0 TO 63
84  AV = "": FOR W = 0 TO 4: UQ = CO(W,Y)
    XOR VO: AP = " ": IF UQ = MX THEN AP =
    BT
85  IF UQ = MX OR UQ = 0 THEN AV = AV +
    STRING$(15,AP): GOTO 88
86  FOR G = 0 TO 14: IF((II^G) AND UQ)<>0
    THEN AV = AV+BT ELSE AV = AV+" "
87  NEXT G
88  NEXT W: LPRINT AV: NEXT Y: GOSUB 900:
    GOSUB 930 ' Olymp: LPRINT ES;"U"
    before NEXT Y
89  FOR I = 1 TO 4: XX(I) = 37: X = 37:
    YY(I) = 32: Y = 32: NEXT I: GOSUB 370:
    GOTO 9
90  '
91  ' <!>: quick draw
92  IF A<>"!" THEN 100 ELSE PRINT @260,"To
    Which?"
93  A0 = INPUT$(1): LA = INSTR(AN,A0):
    PRINT @257, AH
94  IF LA = 0 OR LA>4 THEN A = A0: BEEP:
    GOTO 10
95  X = XX(LA): Y = YY(LA): X0 = X: Y0 =
    Y: PSET(X,Y): GOSUB 370
96  A = INPUT$(1): GOSUB 310: IF P = 0
    THEN BEEP: GOTO 10
97  PSET(X,Y): GOSUB 370
98  FOR J = 1 TO 0 STEP-1:
    LINE(X0,Y0)-(X,Y),J: GOSUB 200: NEXT J
99  PSET(X0,Y0): GOTO 96
100 '
101 ' <9>; <(>: circle BLACK, WHITE
102 Q3 = INSTR("9(0)",A): IF Q3 = 0 THEN
    120 ELSE AX = MID$(BK,Q3,1)
103 LJ = 2: C9 = 1: IF Q3>2 THEN Q3 =
    Q3-2: C9 = 2
105 B4 = "+": IF Q3 = 2 THEN B4 = "-"
106 PRINT @255,AX;" Center,Rim: 1,2,3,4":
    A8 = INPUT$(2): PRINT @255,AH: GOSUB
    290
107 IF IR<>0 THEN IR = 0: GOTO 9
108 N5 = VAL(MID$(A8,1,1)): N6 =
    VAL(MID$(A8,2)): X5 = XX(N5): Y5 =
    YY(N5): X6 = XX(N6)
109 PRINT @255,"Ctr: ";N5;"  Rim: ";N6
110 Y6 = YY(N6): RA = SQR((X6-X5)^2 +
    (Y6-Y5)^2): LP = 7.5*CSNG(RA): XO =
    100
```

```
111 YO = 100: SP = 6.29/CSNG(1+LP): GOSUB
    750: FOR RN = 0 TO LP
112 SR = RN * SP: X = X5 + COS(SR) * RA:
    Y = Y5 + SIN(SR) * RA
113 IF X<0 OR X>74 OR Y<0 OR Y>63 OR (X =
    XO AND Y = YO) THEN 115
114 GOSUB 380: XO = X: YO = Y: IF C9 = 1
    THEN GOSUB 710 ELSE GOSUB 701
115 NEXT: X5 = XX(N5): Y5 = YY(N5): GOSUB
    380: IF C9 = 1 THEN GOSUB 800
119 '
120 ' Screen tone invert
121 IF A<>"8" THEN PRINT @255,AH: BEEP:
    GOTO 9 ELSE FOR Y = 0 TO 63: FOR W =
    0 TO 4
122 CO(W,Y) = CO(W,Y) XOR MX: NEXT W:
    NEXT Y: GOSUB 130: GOTO 9
129 '
130 ' Screen fill
131 LINE(0,0)-(74,63),0,BF: FOR Y = 0 TO
    63: FOR W = 0 TO 4: UQ = CO(W,Y)
133 IF UQ = 0 THEN 137 ELSE X8 = 15*W
134 IF UQ = MX THEN X5 = X8: X6 = X5+14:
    LINE(X5,Y)-(X6,Y),1: GOTO 137
135 FOR G = 0 TO 14: IF((II^G) AND UQ)<>0
    THEN X = X8 + G: PSET(X,Y)
136 NEXT G
137 NEXT W: NEXT Y: LINE(76,0)-(76,63),1:
    X = 37: Y = 32: GOSUB 370: RETURN
139 '
140 ' Calculate G, CX
145 G = X MOD 15: CX = X/15: RETURN
149 '
150 ' For left side of keyboard
151 PSET(X,Y): GOSUB 145: CO(CX,Y) =
    CO(CX,Y) OR II^G: IF B4 = "+" THEN
    155
152 CO(CX,Y) = ((II^G) XOR CO(CX,Y)): FOR
    ZT = 1 TO 50: NEXT ZT: PRESET(X,Y)
155 RETURN
159 '
160 ' Raw bits
161 IF (X>X6 OR X<X5) THEN 180 ELSE GOSUB
    145
165 IF G = 0 AND (X+14) <= X6 THEN 170
167 CO(CX,Y) = (CO(CX,Y) OR (II^G)): X =
    X+1: GOTO 160 ' start at bit >0
170 CO(CX,Y) = MX: X = X+15: GOTO 160 '
    start at bit = 0
180 X = X6: RETURN
184 '
190 PRINT @255,"    Invalid Dup    ":
    BEEP: GOSUB 200: PRINT @255,
    SPACE$(20): RETURN
199 '
200 ' Delay
201 FOR ZT = 1 TO 1000: NEXT: RETURN
210 '
215 ' File-handling utility
216 UF = 0: PRINT @53," Which File?   ":
    AF = "": FOR CL = 93 TO 133 STEP 40
217 PRINT @CL, AH: NEXT CL
218 PRINT @173,"End with   ": PRINT
    @213," .DO       ": LF = 0
220 PRINT @253,AF;"         ": LF =
    LEN(AF)
225 A8 = INKEY$: IF A8 = "" THEN 225
230 IF A8<>CHR$(127) THEN 240 ELSE IF
    LF>0 THEN LF = LF-1: AF =
    LEFT$(AF,LF)
235 GOTO 220
240 AF = AF + A8: LF = LEN(AF): PRINT
    @253,AF;"    "; IF LF<4 THEN 220
242 IF LF>9 THEN BEEP: AF = "": LF = 0:
    UF = 1: GOTO 250
245 A9 = RIGHT$(AF,3): IF A9 = ".DO" OR
    A9 = ".do" THEN 250 ELSE 220
250 FOR CL = 13 TO 253 STEP 40: PRINT
    @CL,AH: NEXT CL: RETURN
289 '
290 ' Check for duplicates
291 FOR I = 1 TO LJ: AP = MID$(A8,I,1):
    LK = INSTR(I,AN,AP)
292 IF LK<1 OR LK>4 THEN IR = 1: BEEP:
    GOTO 296
293 NEXT I: FOR I = 1 TO (LJ-1): FOR LK =
    (I+1) TO (LJ)
294 IF MID$(A8,I,1) = MID$(A8,LK,1) THEN
    IR = 1: GOSUB 190: GOTO 296
295 NEXT LK: NEXT I
296 RETURN
299 '
300 ' Cursor steps
310 P = INSTR(A1,A): IF P = 0 THEN RETURN
315 SI = 15: IF P <= 18 THEN SI = 1 ELSE
    IF P>27 THEN SI = 7
320 B2 = MID$(A2,P,1): IF B2<>"+" THEN
    330
325 Y = Y+SI: IF Y>63 THEN Y = 63
330 IF B2<>"-" THEN 340
335 Y = Y - SI: IF Y<0 THEN Y = 0
340 B3 = MID$(A3,P,1): IF B3<>"+" THEN
    350
345 X = X + SI: IF X>MC THEN X = MC
350 GOSUB 370: IF B3 = "-" THEN 355 ELSE
    360
355 X = X-SI: IF X<0 THEN X = 0
360 RETURN
369 '
370 ' Display coordinates sorted
371 FOR I = 1 TO 4: GR(I) = I + 5*XX(I) +
    400*YY(I): NEXT I
372 FOR I = 1 TO 4: MN = I: FOR J = 1 TO
    4: IF GR(J)<GR(MN) THEN MN = J
373 NEXT I: LS = GR(MN) MOD 5
377 CL = 53+ I*40: PRINT @CL, USING
    "# x## y## !"; MN; XX(LS); YY(LS)
378 CL = CL+ 10: IF Y = YY(LS) AND X =
    XX(LS) THEN PRINT @CL,"*" ELSE PRINT
    @CL," "
379 GR(MN) = MX: NEXT I
```

```
380 PRINT @13,"*": PRINT @15, USING
    "x## y##";X,Y: RETURN
399 '
400 ' Draw Lines
410 DX = X6 - X5: ZX = SGN(DX): DY = Y6 -
    Y5: ZY = SGN(DY): XY = DX*DY: B4 =
    "+"
415 DI = ABS(DX) + ABS(DY)
420 SL = 0: IF DX<>0 AND DY<>0 THEN SL =
    CSNG(DY)/CSNG(DX)
430 IF DI = 0 THEN RETURN ELSE IF
    ABS(DX)>ABS(DY) THEN 600
499 '
500 ' Vertical or steep?
501 IF DX<>0 THEN 520
510 IF DX = 0 THEN LINE(X6,Y5)-(X6,Y6)
520 FOR Y = Y5 TO Y6 STEP ZY: IF DX = 0
    THEN X = X5 ELSE X = X5
    +(CSNG(Y-Y5)/SL)
530 GOSUB 700 ' steep
540 NEXT Y: GOTO 660
599 '
600 ' Horizontal or shallow?
601 IF DY = 0 THEN LINE(X5,Y5)-(X6,Y5),1
620 IF DY = 0 THEN LINE(X5,Y5)-(X6,Y5),1
630 FOR X = X5 TO X6 STEP ZX
640 IF DY = 0 THEN Y = Y5 ELSE Y =
    (SL*(X-X5)) + Y5
645 GOSUB 700
650 NEXT X
660 Y = Y6: X = X6
670 RETURN
699 '
700 ' Build LL, RR
701 GOSUB 145: CO(CX,Y) = CO(CX,Y) OR
    (II^G)
703 IF INSTR("5Ø",A)>0 THEN 705 ELSE IF
    INSTR("%)",A)>0 THEN 707 ELSE 710
705 PSET(X,Y): GOTO 708
707 PRESET(X,Y): CO(CX,Y) = CO(CX,Y) XOR
    (II^G)
708 IF INSTR("9(",A)>0 THEN 710 ELSE
    RETURN
710 IF LL(Y)<0 THEN LL(Y) = X: RR(Y) = X:
    GOTO 730
720 IF X<LL(Y) THEN LL(Y) = X ELSE IF
    X>RR(Y) THEN RR(Y) = X
730 RETURN
749 '
750 ' Initialize LL, RR
751 FOR I = 0 TO 63: LL(I) = -1: RR(I) =
    -1: NEXT I: RETURN
769 '
770 ' Transfer response to array, X#, Y#
771 FOR I = 1 TO 2: K =
    VAL(MID$(A8,I,1)): XX(4+I) = XX(K):
    YY(4+I) = YY(K): NEXT I
780 X5 = XX(5): X6 = XX(6): Y5 = YY(5):
    Y6 = YY(6): RETURN
799 '
800 ' BLACKEN or WHITEN the Shape. Q3: 1
    = BLACK, 2 = WHITE
801 Q1 = MX: Q2 = 1: IF Q3 = 2 THEN Q1 =
    0: Q2 = 0
803 FOR Y = 0 TO 63: IF LL(Y)<0 THEN 840
805 PRESET(76,Y)
810 X5 = LL(Y): X6 = RR(Y)
820 IF X5<>X6 THEN 825
821 X = X5: PSET(X,Y): IF Q3 = 2 THEN
    PRESET(X,Y)
822 GOSUB 145
823 CO(CX,Y) = (CO(CX,Y) OR II^G): IF Q3
    = 2 THEN CO(CX,Y) = (CO(CX,Y) XOR
    II^G)
824 GOTO 840
825 LINE(X5,Y)-(X6,Y),(2-Q3): X = X5
828 IF X>X6 THEN 840 ELSE GOSUB 145
829 IF G = 0 AND ((X+14) <= X6) THEN
    CO(CX,Y) = Q1: X = X + 15: GOTO 828
830 IF X>X6 THEN 840
835 CO(CX,Y) = (CO(CX,Y) OR II^G): IF Q3
    = 2 THEN CO(CX,Y) = (CO(CX,Y) XOR
    II^G)
838 X = X + 1: GOTO 828
840 NEXT Y: IF A = "5" THEN I = VAL(A6)
    ELSE I = VAL(MID$(A8,3,1))
850 X = XX(I): Y = YY(I): GOSUB 370:
    PRINT @255,AH
855 LINE(76,0)-(76,63)
857 RETURN
899 '
900 ' Clear test area
901 FOR CL = 13 TO 253 STEP 40: PRINT
    @CL, SPACE$(26): NEXT CL: CL = 13:
    RETURN
909 '
910 ' Into half-space
911 BT = "*": PI = INSTR("TtIi",BP): IF
    PI<3 THEN 913
912 LPRINT E$;"A";CHR$(6);E$;"2": BT =
    CHR$(254): RETURN ' IBM
913 IF PI = 0 THEN RETURN ELSE IF PI = 1
    THEN BT = CHR$(227)
914 LPRINT CHR$(30): LPRINT CHR$(19):
    LPRINT E$;CHR$(29): LPRINT
    E$;CHR$(28) ' Tandy
915 RETURN
929 '
930 ' To single space
931 IF PI>2 THEN LPRINT E$;"3";CHR$(36):
    RETURN ' I
932 IF PI>0 THEN LPRINT E$; CHR$(54) ' T
933 RETURN
997 '
998 ' Over and Out
999 CLS: CLOSE: MAXFILES = 0: END
```

YOU BE THE EDITOR

Every reader has unique needs. Some seek guidance from reviews of software and peripherals. Others look to a magazine to provide tips to increase their programming proficiency. Accounts of innovative applications attract a big audience, as do articles about other users who find creative ways to use their systems.

You can help tailor **Portable 100** to your unique needs by rating the articles that appeared in this issue. Just circle the numbers indicated below on the Reader Service Card at left. Rate all the articles, some of them or just one. Then detach the card and drop it into the mail — we pay the postage.

The editors will analyze your feedback and respond by commissioning more articles of the most popular types — and fewer of the losers.

Programmed to Learn
An educational tool.
(page 8)
Excellent 160
Good 161
Poor 162

Flick That Switch
CoCo to Model 100.
(page 18)
Excellent 163
Good 164
Poor 165

Scripy
Text processing.
(page 20)
Excellent 166
Good 167
Poor 168

A Font For All Seasons
Special FX.
(page 24)
Excellent 169
Good 170
Poor 171

Command Your Printer
Facilitate spreadsheet output.
(page 33)
Excellent 172
Good 173
Poor 174

Utility Corner
Crazy like a box.
(page 36)
Excellent 175
Good 176
Poor 177

Portable Sketch Pad
Graphics discovery.
(page 40)
Excellent 178
Good 179
Poor 180

Programming Workshop
Labels.
(page 54)
Excellent 181
Good 182
Poor 183

Memory Management
Doggin' hogs.
(page 59)
Excellent 184
Good 185
Poor 186

OVERALL QUALITY OF THIS ISSUE

Excellent 187
Good 188
Poor 189

ADVERTISERS INDEX

Free Information

For free information on products advertised in this issue of Portable 100, locate the Reader Service number corresponding to the advertisement that interests you. Circle the number on the card at left and drop it into the mail. The literature you've requested will be forwarded to you without obligation. Please allow 3-5 weeks for delivery.

Reader Service No.	ADVERTISER	Page Number
62	Covington Group	30
—	80 Micro	48
65	Merritt Computers	35
40	P.C.S.G.	CIII
52	P.C.S.G.	51
71	P.C.S.G.	5
75	P.C.S.G.	10
63	Peptone Software	38
43	PG Design Electronics, Inc.	CIV
—	Portable 100	16
—	Portable Computer Review	17
66	Portable Power	34
50	Portable Programs	58
74	Purple Computing	31
61	SIAS	38
70	Sigea Systems	29
60	Simons Products	30
76	Soundsight	15
73	Soundsight	39
64	Stradivarius Software	34
67	SunBelt	35
42	Tandy	7
53	Touchbase Systems, Inc.	57
41	Traveling Software	CII
77	Traveling Software	1
72	Traveling Software	23
78	ULTRASOFT	43

SAVE 48% RIGHT NOW!

"I have found something in almost every issue that has been worth the price of the subscription. I am amazed at the broad spectrum of the technical articles you publish, with ample material for both novice and advanced programmer."

Roger L. *Holstege*. . .MD

And with 6 to 12 new programs for your Tandy computer in every issue, you too will find something each month that's worth the low cost of your subscription! Just type them in, and they're yours. Spreadsheets, word processors, all kinds of home, business and personal applications, plus helpful **utilities** that make all your programs easier to write, debug and use.

VALUABLE HARDWARE PROJECTS

Want more value? How about *80 Micro*'s money saving **hardware projects** you build yourself. . . or our frequent **patches and enhancements** that make commercial software programs even better.

PROGRAMMING TECHNIQUES & TUTORIALS

You'll add to the value of your computer and get more out of every hour you spend computing with expert guidance from *80 Micro*'s **programming techniques** that help you streamline and improve programs. In addition, *80 Micro*'s step-by-step **tutorials** keep you ahead of today's fast-moving technologies like nothing else.

HARDWARE & SOFTWARE REVIEWS

Avoid even one chancey hardware purchase, and you've saved enough to repay years of *80 Micro* subscriptions! You'll save yourself a bunch of time and hassle, too, with *80 Micro* checking out new software for you. Zero in on programs and products you want and skip the clunkers.

► For even quicker service, ☎ **CALL TOLL FREE 1-800-258-5473** (in NH, dial 1-924-9471) and charge it to your credit card!

THE MOST WIDELY READ MAGAZINE IN ITS FIELD

80 Micro is the most widely read magazine serving Tandy users today, with the largest editorial staff in its field. It's the oldest and most authoritative, too, so you know you can trust the help and advice you find in every issue.

START SAVING NOW WITH 48% OFF THE COVER PRICE!

Yes it's true. You don't even have to wait for your first issue to start saving with *80 Micro*! Just use the coupon below or return the postpaid card opposite and you'll start receiving *80 Micro* every month at a full **48% off** the Newsstand Rate. ***If you're ever dissatisfied, for any reason, you'll be reimbursed for all un-delivered issues. Guaranteed!*** So do it now. And get *80 Micro*'s value working for you, *every* month from now on!

* TRS-80 is a trademark of Radio Shack, a division of Tandy Corp.

YES, I want to add the value of *80 Micro* every issue. Enter my no-risk subscription for 12 monthly issues at the Introductory Rate of just $24.97! I save a full **48% off** the newsstand price!

☐ Payment enclosed ☐ Bill me

Name (Please Print)

Address

City State Zip

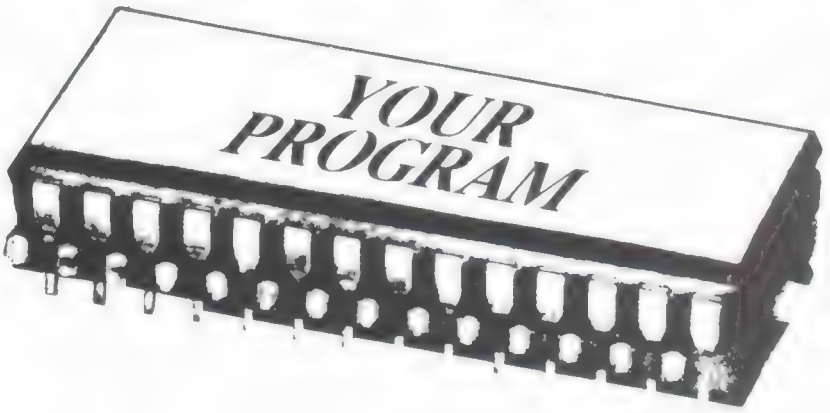
Canada and Mexico, $27.97, Foreign surface $44.97, 1 year only. Foreign airmail, $79.97. US funds drawn on US bank. Please allow 6–8 weeks for delivery.

378B8

80 Micro • Box 981 • Farmingdale, NY 11737

```
    :PRINT" Printer not ready . . Hit any key";
    :R$=INPUT$(1)
    :GOTO 1449
 1415 GOSUB 1050
    :LINE INPUT "STRING TO PRINT:";TS$
    :IF TS$=""THEN 1449
 1420 INPUT "PROPORTIONAL MODE";R$
    :LPRINT ES$;"p";
    :IF INSTR(1, "Yy", R$) THEN LPRINT CHR$(49);
     ELSE LPRINT CHR$(48);
 1425 LPRINT TS$
 1449 CLS
    :RETURN
 1500 REM KLONE A CHARACTER
 1505 NE=1
    :CLS
    :PRINT "  * * COPY A CHARACTER * *"
    :GOSUB 1050
    :A1=0
    :A2=0
    :INPUT "SOURCE ASCII";A1
    :INPUT "DEST ASCII";A2
    :IF A1=0 OR A2=0 THEN 1549
 1508 PRINT "OK? (Y/N)";
    :IF INPUT$(1) <> "Y" THEN PRINT
     ". . CANCELLED. ."
    :RETURN
 1510 A1=A1-33
    :A2=A2-33
    :FOR QQ=0 TO 11
    :CD(A2, QQ)=CD(A1, QQ)
    :NEXT QQ
    :RETURN
 1549 CLS
    :RETURN
 1550 REM BLANK A CHARACTER--ENTRY/CHAR
 1552 IF NE THEN RETURN
 1555 FOR CCUR=1 TO 11
    :CD(CHAR, CCUR)=0
    :NEXT
    :GOSUB 1790
    :FL=1
    :RETURN
 1600 REM ERROR HAND LING
 1605 IF ERR=7 AND ERL=1110 THEN BEEP
    :PRINT". . MEMORY OVERFLOW. .HIT ANY KEY"
     ELSE 1610
 1606 IF INSTR(1, F$, ".DO")=0 THEN F$=F$+".DO"
 1608 INPUT$(1)
    :KILL F$
    :RESUME 100 'GOSUB STACK IS LOST OR MEMORY
     ERROR
 1610 IF ERR=9 OR (ERR=5 AND (ERL=1325 OR
     ERL=1510)) THEN BEEP
    :PRINT "BAD INPUT VALUE . . hit any key"
    :R$=INPUT$(1)
    :RESUME 1549
 1615 IF ERL<>1110 AND ERL<>1108 THEN 1625 ELSE IF
     ERR>58 THEN PRINT". .DISK ERROR"
 1616 IF ERR=64 THEN PRINT ". .DISK FULL"
 1617 IF ERR=63 THEN PRINT". .DISK WRITE
     PROTECTED"
 1619 IF ERR=59 THEN PRINT". .DRIVE NOT READY"
 1620 BEEP
    :PRINT ". . hit any key"
    :R$=INPUT$(1)
    :RESUME 1115
 1625 IF ERL<>1206 THEN 1690
 1627 IF ERR=52 THEN BEEP
    :PRINT". .FILE NOT FOUND"
 1628 IF ERR=59 THEN PRINT ". .DRIVE NOT READY"
 1629 BEEP
    :PRINT "FILE ERROR. . .HIT ANY KEY"
    :R$=INPUT$(1)
    :RESUME 1215
 1690 PRINT "ERR:";ERR;" IN LINE:";ERL
    :STOP
 1692 RESUME NEXT
 1700 REM SHIFT CONTROLLER--ENTRY/CHAR=CURRENT
     ASCII
 1704 IF NE THEN RETURN ELSE PRINT @200,
     "R/L/U/D/[enter]?";
    :R$=INPUT$(1)
    :GOSUB 1790
    :FL=1
 1705 ON INSTR(1, CHR$(13)+"RDUL", R$) GOSUB 1715,
     1720, 1730, 1740, 1750
 1710 PRINT @200, "                    ";
    :GOSUB 1000
    :RCUR=1
    :CCUR=1
    :RETURN
 1715 RETURN
 1720 FOR QQ=10 TO 1 STEP -1
    :CD(CHAR, QQ+1)=CD(CHAR, QQ)
    :NEXT QQ
    :CD(CHAR, 1)=0
    :RETURN 'RIGHT SHIFT
 1730 FOR QQ=1 TO 11
    :CD(CHAR, QQ)=CD(CHAR, QQ)\2
    :NEXT QQ
    :RETURN 'SHIFT DOWN
 1740 FOR QQ=1 TO 11
    :CD(CHAR, QQ)=(CD(CHAR, QQ)*2) AND 255
    :NEXT QQ
    :RETURN 'SHIFT UP
 1750 FOR QQ=2 TO 11
    :CD(CHAR, QQ-1)=CD(CHAR, QQ)
    :NEXT QQ
    :CD(CHAR, 11)=0
    :RETURN 'LEFT SHIFT
 1790 FOR QQ=31 TO 271 STEP 40
    :PRINT @QQ, "        ";
    :NEXT QQ
    :GOSUB 300
    :RCUR=1
    :CCUR=1
    :SW=1
    :GOSUB 620
    :GOSUB 630
    :RETURN 'DRAW NEW GRID
```

Your own Program on a ROM



8K or 32K
Your BASIC or machine code program on a custom ROM ready to snap into your Model 100, Tandy 102 or Tandy 200.
You can't tell your program from a built-in. Starting at $50.00 each. Minimum order of 10. Quantity pricing as low as $15^{00}

PORTABLE COMPUTER SUPPORT GROUP

11035 Harry Hines Blvd. No. 207, Dallas, Tx. 75229

1-214-351-0564

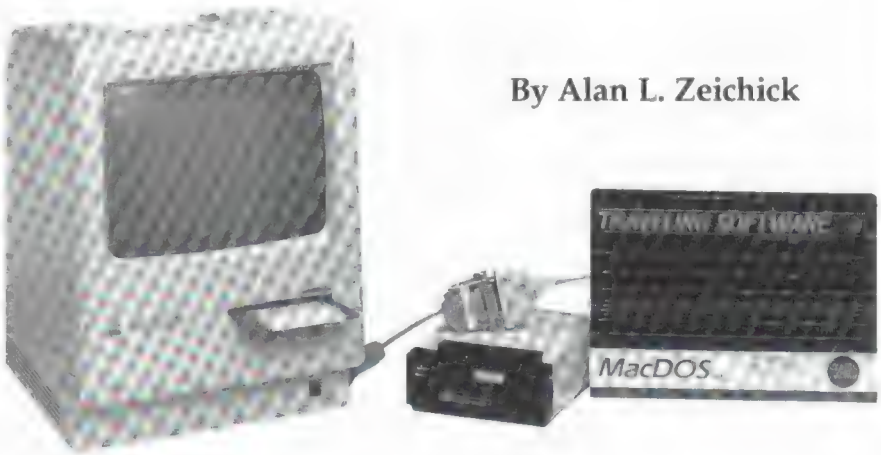
VISA / MASTER CARD / C.O.D.

Circle 52 on Reader Service Card

MacLaptop, Here We Come

MacDOS lets your Apple Macintosh read and write to the Tandy Portable Disk Drive.

By Alan L. Zeichick



Attention, Apple Macintosh owners: Your Mac can read, write and initialize Tandy Portable Disk Drive disks. Not directly, mind you; you can't stick PDD disks into the Mac's built-in 3.5-inch disk drive. But you *can* attach the Portable Disk Drive to the Mac and use it as full-featured peripheral.

The product that bridges the Mac gap is MacDOS, from Bothell, Wash.-based Traveling Software. According to Traveling Software president Mark Eppley, the $94.95 MacDOS gives the Mac total control over the Portable Disk Drive.

HOOKING IT UP

Opening the MacDOS package reveals a thick brown-and-white manual, a blue 3.5-inch Macintosh disk, and a short, six-inch grey cable. This cable, labeled "MacDOS Cable Converter," has a *D*-shaped 25-pin (DB-25) female connector at one end, a smaller DB-9 male connector at the other. The flat black cable that comes with your Tandy Portable Disk Drive connector plugs into the DB-25 end of the MacDOS cable; mine fits snugly, but the locking screws on the MacDOS cable won't reach far enough to form a mechanical bond with the PDD cable. Since the PDD won't usually be permanently attached to the Mac, though, that's not a problem.

The DB-9 end of the MacDOS cable fits into the *modem* connector on the back of the original Macintosh or Mac 512 (the icon on the back of the Mac is a little telephone handset). The connector also fits into the *printer* port, but that's not advisable. The Mac's modem port has a higher internal interrupt priority than the printer port, so most communication programs — MacDOS included — use the faster, more reliable modem port.

Macintosh Plus and Mac SE owners, please note: Your computer doesn't use DB-9 connectors on the printer and modem ports. Instead, you'll find a small, round *Mini-8* jack. You'll need to purchase an adapter cable from your Apple dealer — it's well worth acquiring, since you may want it for other Macintosh accessories in the future.

The cable is called, appropriately enough, the Mini-8-to-DB-9 adapter; Apple Computer's part number is 590-0341. Be careful when you set up your cables, since you're going to have a long expanse of cables: Portable Disk Drive to black flat cable to MacDOS cable to adapter to Macintosh. Needless to say, you shouldn't wiggle the cables during a data transfer.

Remember those annoying DIP switches on the Portable Disk Drive? Those must be set, as appropriate. The newer PDD-2 and the Brother FB-100 only have one data transfer rate — no problem there. On the original PDD, and on the Purple Computing drive, set the switches on, for a rapid 19,200 bits per second (bps).

You should have no trouble connecting the PDD to the Macintosh. The steps are simple, and the MacDOS manual devotes five well-written pages to the subject, complete with informative MacPaint-type drawing of cables, connectors, and the back of the Mac. I connected an original PDD to both an original Macintosh and a Macintosh SE, with no problems with the cables or the physical connection.

LOAD IT AGAIN, SAM

Next, to software. If you have a hard disk attached to your Macintosh, copy the data from the MacDOS disk onto the disk, preferably into its own folder. If you don't have a hard disk, of course, you should make a back-up copy of the MacDOS disk. It's not copy-protected.

Click on MacDOS's icon. You'll find a true Macintosh-style file display window, complete with three pull-down menus: *File, Edit* and *PDisk*.

The PDisk pull-down options are perhaps the most useful. These include:

- Catalog: Displays the names and sizes of the Portable Disk Drive files. This option must be selected before the catalog will appear, since your PDD might be turned off or not contain a disk when you start running MacDOS. For the mouse-haters, Traveling Software implemented key combinations for the menu choices; this one's Control-D.

If the Portable Disk Drive is turned

off, doesn't contain a usable disk, or contains an unformatted disk, an appropriate error dialog box is displayed.

- Open: Opens as many as four windows into the Mac's other drives, so that you can look for files to be transferred to the PDD, or for places to put PDD files. These windows can show the contents of any hard disk or floppy folder. I found *four* windows excessive, but during complex file-transfer operations they might be useful.
- Save and Save As: Any file shown in the Macintosh folder windows can be saved to the PDD by single-clicking on it, and then selecting the Save or Save As options. Since the PDD has a six-character name length limit — and the Mac does not — Save As provides the opportunity to pick a shorter, more meaningful name. And since PDD files must be tagged as document (.DO), BASIC (.BA) or machine-language (.CO), you can click on the appropriate file extension. (If your Mac file already has one of these file extensions, you don't need to re-select one manually). Be sure to pick the right file extension! If you save a BASIC file to the PDD as a .CO file, and try to execute it on the Model 100 (or other similar laptop), you're in for a cold start.
- Kill: This option deletes PDD files.
- PDisk to Mac: Select this option to move a file from the PDD to the Macintosh. A dialog box will help you select a destination on one of the Mac's disk drives.
- Copy All: This option copies the entire contents of the Portable Disk Drive to the Mac. Use this option when you're using the Mac's hard disk, for example, as short or long-term storage for your Model 100. Unfortunately, there's no option to copy all of a Macintosh folder to Portable Disk Drive in one operation; this would be handy when trying to mass-produce PDD disks.
- Init: Initializes, or formats, a Portable Disk Drive disk. The operation takes about 90 seconds. I tried formatting a disk that was known to be defective, and an error-message dialog box gave me clear indication that my disk was bad.
- Duplicate: Uses the Mac's memory to duplicate, or create a back-up copy, of a Portable Disk Drive disk. Both Macintosh computers that I used had sufficient memory to duplicate a fully-laden disk without multiple disk swaps, but the MacDOS manuals warns that some Macs may require several disk swaps.
- Choose Bank: The newer Portable Disk Drive 2, which uses double-sided disks and has a disk capacity of 200K, uses a two-bank file directory in order to maintain compatibility with the older single-sided PDD. Use this option to switch for one directory bank to another. And yes, the Copy All option above only works on current PDD-2 bank.

EVEN MORE OPTIONS

I found that my Macintosh text editors — Microsoft Word, Ready, Set, Go! and MockWrite — were perfectly able to read and edit PDD files, once those files were copied onto a regular Mac floppy or hard disk. And, files created by those editors could be copied back to the Portable Disk Drive, if the files were saved in ASCII format. That means, for example, that a native Microsoft World document, full of type style and sizes, can't be dumped "as is" to the PDD and from there to the Model 100; you'll have to convert the Word-format file to straight ASCII text. But that's only to be expected.

MacDOS includes a handy text editor that lets you modify PDD disk files direct, *without* copying them to the Mac. The text editor is well-suited to its mission, with text insertion and deletion, cut-and-paste, and the ability to move text to and from the Macintosh clipboard. Of course, if more editing power is required, you can copy the PDD file to a Mac hard or floppy disk and use a regular word processing program.

I found all of the various options and abilities built into MacDOS sufficient for most applications; the one weakness being the inability to copy an entire Macintosh folder to the PDD in one operation.

Toward the back of the MacDOS manual, I found an interesting feature: Printing the PDD's disk catalog, or copying the catalog to a disk file. There's no automatic catalog-printing menu option, but the catalog text can be copied-and-pasted into a new file, which can be printed from a menu. Cumbersome, but it *does* get the job done.

I had one scary moment: During one test, I formatted (initialized) a blank disk. Then I tried duplicating a disk, first selecting *duplicate* from the pull-down menu, then inserting the source and destination disks as directed by dialog boxes. At the conclusion of the test, I had the destination disk in the Portable Disk Drive — but the PDD catalog box on the screen didn't display any file names.

Was my copy successful? It was. I had forgotten that I must explicitly select *catalog* from the PDisk menu; MacDOS was still showing the catalog from my earlier newly-formatted disk.

Having passed all of its MacDOS tests, I reconnected the Portable Disk Drive to its familiar Tandy 102. The laptop had no difficulty using the newly initialized disk, or reading any of the files copied from the Mac. That's not really surprising, since the Macintosh surely created the files using the same disk-drive commands that the laptop does.

PROFESSIONAL JOB

With the sole exception of the interesting screen display within the PDD catalog window, where some characters are in one screen font and others are in another, the MacDOS product fits well into the look-and-feel of Apple Macintosh applications. The 60-page manual is well-written, and most important, the product does the job.

If you own an Apple Macintosh and a Portable Disk Drive, you might wish to consider purchasing MacDOS if you anticipate using your laptop computer to feed information into the Mac. MacDOS is cleaner and simpler than using a null-modem and communication software; and besides, unless you have an XMODEM-compatible telecommunications product for the Model 100 such as Sigea System's X-TEL, you can't transmit machine-language programs over a modem anyway. And, if your budget can stand owning *two* PDDs, a Model 100, Tandy 102 or Tandy 200 in the field can mail data disks back to the home office.

Macintoshes with hard disks can also serve as convenient file repositories for Tandy laptops. It's easy to set up a 3.5-inch disk with exactly the file necessary from a large, multi-megabyte library of documents, BASIC programs and machine-language utilities. This I find the most useful application; I've been using the MS-DOS equivalent of MacDOS, LapDOS, on my hard-disk equipped IBM PC/AT to archive Model 100 files for *Portable 100* for quite some time.

Conversely, Portable Disk Drive disks aren't very expensive, and if you rarely feel a need to share more than the occasional document file between your Macintosh and Model 100, MacDOS may be overkill. But if you use your laptop as a true portable computer, and depend on your Mac for use in the office, MacDOS could make your life easier and more productive. □

One, Two, Three Across, Four Across Rock

Print more than just simple mailing labels on your laptop

By Alan L. Zeichick

The saying goes that necessity is the mother of invention — and this article's no exception. A few weeks ago a young lady from St. Louis called me with a dilemma. Her situation: She owns a Model 100 and an Epson FX-80 printer, and wants to print mailing labels. The problem: The built-in pin feed on the Epson printer wouldn't adjust narrow enough to fit the 3.75-inch standard mailing label carrier —wouldn't even come close.

Her choices, therefore, were either to purchase an expensive external tractor feed for the printer, or use two-across (or two-up) labels on a wider carrier. She didn't want to purchase any more accessories, didn't know how to print two-up labels from BASIC — and didn't relish the idea of throwing half of her labels away unused. Could I help?

Well, I could. Yours truly hadn't written a program recently, and was getting itchy — and a general-purpose mailing-label generator for the Model 100, Tandy 102 and Tandy 200 seemed like a good project.

LABEL-MASTER

The approximately 3,500-byte LABELS.BA is a truly generic label program, able to produce labels in any height, width and number across each page. But before we examine the program, let's take a look at its input.

The text, or data, for the labels is resident within a RAM .DO document file. (If you have one of the disk-drive operating systems that allows you to access disk files with a file prefix letter, you can store data on disk without modifying LABELS.)

The format is simple: Each line contains one record. For example, Suzie's name and address are on line, John's on the next, etc. Each separate bit of information within a record — what's called a *field* — is held apart with a *field separator*, initially defined as a colon. There should not be a space after the colon, unless you want your fields to start with a leading blank. A record, therefore, might look like *Suzie Smith:Smith & Associates:100 Main Street: Anytown, USA 12345.*

You can place as many fields on a line as you wish, with only one restriction: The total length of each record should not exceed 255 characters. The number of fields actually printed on a mailing label depends on the size of the label: If you've told LABELS that the label is six lines long, the first six fields will be printed. If there are more than six fields, the remaining fields will be ignored, which makes them ideal for storing information such as telephone numbers or non-printing comments.

Now that the data file's been created, let's print some labels! Select LABELS.BA from the main menu (assuming that the program's been typed in). You'll be greeted with the prompt, "Label data file?" Enter the name of the data file; you can leave off the .DO file extension if you wish.

The next question is, "Output device (D=LPT:)?" It's asking where to send the labels, with a default, or an assumed option of LPT:, or the parallel printer port. To use the default at this or any other question, simply press Enter; otherwise, type in a new choice and *then* press Enter. The possible answers to this choice are Model 100 device and file names, such as LCD: for the display, MDM: or COM: for the modem or serial ports, or FRED.DO for a RAM document file. Look in your laptop's manual under the BASIC OPEN command for a more complete description of device names.

The technique of opening the parallel printer for output and then printing to it as a numbered device is very efficient from a programming and debugging viewpoint. It's easier to understand — and modify — than the LPRINT statement (although, granted, it's a bit slower at execution time and requires extra RAM for program storage). You can route test data to the screen by entering LCD: as the output device, or to a RAM or disk file for closer examination. I've seen many programs which ask questions like, "Send output to the screen or to disk?" and then use GOTO and IF-THEN statements to choose between LPRINT and PRINT # statements. Well, I think that's pretty inefficient. Try using LPT: or LCD: as a device in your OPEN statement. You'll like the flexibility!

MOVING RIGHT ALONG

After we've selected our input and output devices, we're presented with a four-option menu:

```
Change parameter values
Print an alignment pattern
Start print labels
Main menu
```

Selecting C allows you to modify the program's built-in defaults. The default settings are for one-up labels which start printing one character from the printer's left margin, with six print lines per label and with each label 35 characters wide. These defaults are probably *not* correct for you, depending on your printer's physical and logical setup — for example, if you print 12 characters per inch (12 cpi, or *elite*), you can get 20 percent more on each line than if you print at 10 cpi (or *pica*). And some printers can print at 17 cpi!

You can also have more usable lines per label if you use larger labels, or if you tell your printer to print lines closer together. Normal line spacing is six lines vertically per inch (lpi), so the industry-standard one-inch label

prints five lines (plus one to separate each label, for a total of six). But if you change the line spacing, you can legibly print at eight lpi, which gives you seven usable text lines on a single one-inch label.

Adjust the absolute left margin so that each line of text prints a short distance — perhaps one-eighth inch — from the left side of the first label. It's important to set the left margin properly before adjusting the label width value.

If you use two, three or four-up labels, you'll need to adjust the number of characters per label accordingly. For example, if the left edges of your three-up labels are 3.75 inches apart, and you're printing at 12 cpi, you'll want to set the label width at 45 characters. Be sure to measure from the edge of one label to the same edge of the next to properly account for the space between the labels, or *gutter*.

After you try a set of parameters — and sometimes it ends up being trial-and-error — use the alignment-pattern option on the LABELS menu. This prints a fake mailing label comprised of all *x*'s. You can keep printing this over and over while adjusting the physical labels or the program's parameters until the label looks just right.

After the parameters are set, write them down! You won't want to have to figure them out next time you use those labels. And, if these are the values that you're most likely to use for *all* labels, consider modifying the default-assignment section of the LABELS program, lines 195 through 235. You can also change the data-file field separator in line 230 from a colon to any ASCII character that you're not going to use inside a label text. A Grph character is a good candidate for field separator.

Now, after the alignment pattern has printed perfectly, press S to start printing actual labels. And there you go!

BEHIND THE LABEL

Let's take a quick look behind the scenes. LABELS is written in a semi-structured format, using subroutines to improve program flow.

RAM-conscious readers, take note: If you're watching every byte, you can ignore all of the comment lines while typing in the LABELS listing — none of the comments are referenced in GOTO or GOSUB statements. Better still, type the program in as-is, correct any typos, and then use a program squasher like

```
100 ' Program to print multi-up labels
105 ' Alan L. Zeichick, Portable 100
110 '
115 ' Set up default values
120 '
125 GOSUB 180
130 '
135 ' Read in user values, execute
140 '
145 GOSUB 270
150 '
155 MENU
160 '
165 ' ================================
170 ' Default values
175 '
180 DEFINT A-Z
185 DEFSNG W
190 '
195 UP = 1                  ' Number of labels wide
200 LM = 1                  ' Initial left margin
205 CL = 35                 ' Chars per label
210 LN = 6                  ' Lines per label
215 SP$ = ":"               ' Field separator
220 RV$ = CHR$(27) + "p"    ' Reverse video
225 PV$ = CHR$(27) + "q"    ' Plain video
230 PF$ = "LPT:"            ' Output device
235 DM = 0                  ' Array dimensioned?
240 '
245 RETURN
250 '
255 ' ================================
260 ' Talk to the user
265 '
270 CLS
275 PRINT RV$ LEFT$("Portable 100
    Label-Master"+STRING$(40,32), 40) PV$
280 LINE INPUT "Label data file? "; DF$
285 IF INSTR(DF$,".") = 0 THEN DF$ = DF$ + ".DO"
290 ON ERROR GOTO 850
295 OPEN DF$ FOR INPUT AS 1
300 ON ERROR GOTO 0
305 '
310 CLS
315 PRINT "Output device (D=" PF$ ")";
320 INPUT PF$
325 ON ERROR GOTO 850
330 OPEN PF$ FOR OUTPUT AS 2
335 ON ERROR GOTO 0
340 '
345 CLS
350 PRINT "Do you wish to:
355 PRINT "   " RV$ "C" PV$ " Change parameter
    values,"
360 PRINT "   " RV$ "P" PV$ " Print an alignment
```

the one supplied with the *Cleuseau* ROM to save RAM and minimize program execution time.

The first executable statements assign several program-wide defaults, such as the length of labels and field separator character. Next, the menu. Note that you don't have to bother checking or changing defaults, or even print alignment patterns, but can simply print labels. That's handy when you want to print labels from several data files consecutively, and *know* that everything's lined up properly between runs.

Each menu option, other than Main Menu, calls a subroutine. The first of these lets you override the defaults. Programming note: On the Model 100, if a program INPUTs a variable, and the user simply presses Enter, the variable's value is left unchanged. That's very convenient, as you can see. But other BASICs may treat that situation differently. On the IBM PC, for example, the GW-BASIC interpreter would set numeric variables to zero, and string variables to the null string (""), losing their previous values.

The next menu option prints the alignment pattern. Unlike many other label programs that I've seen, the alignment pattern is *complete*, running the full width of each label, and printing for every label line except for the last, which is assumed to be a blank line between rows. And if the alignment isn't correct, the user can manually adjust the printer carriage or tractor feed and retry, or run through the configuration again.

By the way, if you're using labels that *don't* have a blank line between each row, don't worry. If you've told the program that you have six lines per label, and you've entered six data fields per record, LABELS will print all six lines. It's only on the alignment pattern that the program assumes that you have a blank line between rows.

The largest block of lines are concerned with reading data from the label data file one label-row at a time (lines 660 through 715), and then printing that data onto the printer (745 through 775). Data is are read and labels are printed until the program reaches the end-of-file; at that point, the program prints any odd labels remaining and closes the input data file.

An extra feature: If your label data file contains a line with the word "SKIP,", the program will neatly skip a row of labels. Why? Well, let's just say, "For future use."

```
    pattern,"
365 PRINT "   " RV$ "S" PV$ " Start printing labels,
    or"
370 PRINT "   " RV$ "M" PV$ " Main Menu?"
375 '
380 IN$ = INPUT$(1)
385 IF IN$ = "C" OR IN$ = "c" THEN GOSUB 435
    :GOTO 345
390 IF IN$ = "P" OR IN$ = "p" THEN GOSUB 525
    :GOTO 345
395 IF IN$ = "S" OR IN$ = "s" THEN GOSUB 610
    :GOTO 345
400 IF IN$ = "M" OR IN$ = "m" OR IN$ = CHR$(27) THEN
    RETURN
405 '
410 GOTO 380
415 '
420 ' ================================
425 ' Get user defaults
430 '
435 CLS
440 PRINT RV$ "Press return to keep defaults:" PV$
445 PRINT "Absolute left margin (D=" STR$(LM) ")";
450 INPUT LM
455 PRINT "Chars per label (D=" STR$(CL) ")";
460 INPUT CL
465 IF DM THEN RETURN
470 '
475 PRINT "Labels across (D=" STR$(UP) ")";
480 INPUT UP
485 PRINT "Lines per label (D=" STR$(LN) ")";
490 INPUT LN
495 '
500 RETURN
505 '
510 ' ================================
515 ' Print an alignment pattern
520 '
525 CLS
530 PRINT RV$ "Press any key when the printer's ready"
    PV$ INPUT$(1)
535 '
540 FOR J = 0 TO LN - 2
545     PRINT #2, STRING$(LM, 32);
550     FOR I = 0 TO UP - 1
555         PRINT #2, STRING$(CL - 2, 88) + "  ";
560         NEXT I
565     PRINT #2, ""
570     NEXT J
575 PRINT #2, ""
580 '
585 RETURN
590 '
595 ' ================================
600 ' Reading and printing labels
```

```
605 '
610 IF NOT DM THEN DIM DA$(UP, LN)
    : DM = -1
615 '
620 PR$ = STRING$(CL, 32)
625 SK = 0
630 EF = 0
635 CLS
640 PRINT RV$ "Press any key when the printer's ready"
    PV$ INPUT$(1)
645 '
650 ' Fill the array from the file
655 '
660 FOR I = 0 TO UP - 1
665     ON ERROR GOTO 880
670     IF NOT EF AND NOT SK THEN LINE INPUT #1, ST$
        ELSE ST$ = ""
675     ON ERROR GOTO 0
680     IF ST$ = "SKIP" THEN ST$ = ""
        :SK = -1
685     ST$ = ST$ + STRING$(LN, SP$)
690     FOR J = 0 TO LN - 1
695         K = INSTR(ST$, SP$)
700         IF K > 1 THEN DA$(I, J) = LEFT$(ST$, K-1)
            ELSE DA$(I, J) = ""
705         ST$ = MID$(ST$, K+1)
710         NEXT J
715     NEXT I
720 '
725 ST$ = ""
730 '
735 ' Print the array onto the labels
740 '
745 FOR J = 0 TO LN - 1
750     PRINT #2, STRING$(LM, 32);
755     FOR I = 0 TO UP - 1
760         PRINT #2, LEFT$(DA$(I, J)+PR$, CL);
765         NEXT I
770     PRINT #2, ""
775     NEXT J
780 '
785 ' If skip, then do so; then get more data
790 '
795 IF SK THEN FOR J = 0 TO LN - 1
    :PRINT #2, ""
    : NEXT J
    :SK = 0
800 IF NOT EF GOTO 660
805 CLOSE 1
810 OPEN DF$ FOR INPUT AS 1
815 RETURN
820 '
825 ' ================================
830 ' Error handling
835 '
```

BACK TO THE MENU

From there, back to the LABELS menu. You can change the alignment and print the labels or alignment pattern again — useful for making multiple copies of data file. However, you can't change certain parameters once you've printed labels, notably the number of labels across and the number of lines per label. That's because of the dimensioning of the DA$ array, and the fact that Model 100 BASIC won't let you de-allocate array space. So, to change those parameters, or to change data files, exit back to the laptop main menu, and re-execute LABELS.

LABELS, as written and presented, is ready to use. But there are some enhancements that you might wish to make. One is to have the program automatically print any printer setup codes. You might wish to print CHR$(27); CHR$(77) to set an Epson printer to 12 cps (*elite*) mode, or CHR$(27); CHR$(48) to set line spacing to eight lines per inch.

You might also wish to write separate programs to manipulate the .DO data file — I did. These might include a general-purpose utility to sort the data file by designated fields or print reports based on the contents of the data file. □

Variable List for LABELS.BA

Variable	Usage
CL	Character width of each label
DA$(UP, LN)	Data array for label information
DF$	Input data file name
DM	Boolean test for DA$ prior dimensioning
EF	Boolean end-of-file
I	Loop counter for labels across a page
IN$	Characters read from keyboard
J	Loop counter for label lines
K	Position of SP$ within ST$
LM	Absolute left page margin
LN	Height of each label, in lines
PF$	Output file name
PR$	Character string made up of blanks
PV$	Escape sequence for plain video
RV$	Escape sequence for inverse video
SK	Used in skipping rows of labels
SP$	Data file field separator
ST$	Character string used read from data file
UP	Number of labels across a page

```
840 ' No such data file
845 '
850 RESUME 855
855 PRINT RV$ "Error:" PV$ " " DF$ " not found."
    INPUT$(1)
860 MENU
865 '
870 ' End of input data
875 '
880 EF = -1
885 ST$ = ""
890 RESUME NEXT
```

Model 100 Memory Management

How to get a handle on those memory hogs.

by Thomas L. Quindry

The Model 100 and Tandy 102 computers offer a lot of useful utility for the writer on the go. Many times, though, when away from the office, you need just that little bit of extra memory to finish what you are doing or writing. To get that extra memory, you will have to find out just where the memory hogs are in your computer and perhaps kill a file or two to make room for your current chore.

With this purpose in mind, I wrote the program, LENGTH.CO. This little machine language program is only 198 bytes long. It packs a big wallop for its size by letting you know everything you need to know about memory usage in your computer. Besides the usual fare of telling you the size of each program, LENGTH also gives you other needed information to determine where you can recover memory, and it gives you the amount of free memory you have as well.

Other areas besides programs can eat up memory also and they are much easier to free up. These are the Free BASIC area, the Paste Buffer, Memory protected by the HIMEM pointer, and the MAXFILES buffer.

HOW TO USE

LENGTH is easy to use. Just run the program and read the names of your programs and the other factors as described. Filenames are followed by the number of bytes that make up the file. In the file listing displayed, the type of file—.BA, .DO, .CO—is not given in order to be able to list all information on one screen. With only twenty filenames possible, it should be easy for you to remember the kinds of files you have. Hidden files are not displayed, as I make the assumption that they are hidden so they won't be killed. After reviewing the display, hitting any key

TABLE 1. System information and ROM calls used by LENGTH.CO.

Address	Description
0000H	A jump to here exits from program and returns to MENU
12CBH	Wait for keyboard character and returns it in A register
20D5H	Used to check directory entries - increments HL by 11 bytes
344AH	Call to here will give address (in DE) of program name pointed to by HL register
39D4H	Call to here converts value in HL register to decimal and displays number
4222H	Prints Carriage Return and Linefeed on screen
5AE3H	Loads HL with the value at memory location HL
6BF4H	The string "BASIC" starts here - This is pointed to when the word "BASIC" is displayed
7EACH	Call to here will print the number of bytes of free memory
F5F4H	This byte contains the value of HIMEM
F639H	This byte contains the row number where the cursor is
F63AH	This byte contains the column number where the cursor is
F685H	Beginning of keyboard buffer - LENGTH.CO starts 36 bytes above here
F99AH	This byte contains a pointer to the start of FREE BASIC
F9A5H	This byte contains a pointer to the start of the Paste Buffer
F9BAH	Points to first user file in directory - I point to 11 bytes before this and increment by 11 to read directory
FC82H	This byte contains the number of MAXFILE buffers reserved

will return you to the MENU screen.

BASIC AND THE FREE BASIC BUFFER

The Free BASIC area gets filled when you write your own program into the Model 100 or 102 and then forget to save it to a file. When you go back to the MENU screen, you don't lose your program in Free BASIC. Instead, it is saved there and is available every time you go directly back to BASIC. You can run the program to your heart's content by giving the run command; just as if you loaded it by filename into BASIC. You can clear this area by using the NEW command. The minimum value for Free BASIC is 2. Free BASIC, like any other BASIC file, when empty has two bytes of zero values.

TEXT AND THE PASTE BUFFER

The Paste Buffer is used while in the text mode. When you Select text using the F7 function key and then Cut (F6) or Copy (F5) the text, the selected text goes into the Paste Buffer. The Paste key will copy this text from the Paste Buffer as many times as you want; even to a different file, or to BASIC without error if every statement ending in a carriage return is preceded by a line number. Text in the Paste Buffer remains there until you again use the Select key or else when you use the CLEAR command from BASIC.

The easiest way to clear the Paste Buffer is to hit the F7 key and then immediately the F6 or F5 key while in the text mode. The minimum value for the Paste Buffer is one. The Paste Buffer, like any other Text file, when empty has one byte of value 1AH.

MACHINE LANGUAGE FILES

The calculation of length of a machine language file is subject to interpretation. In its storage location, a machine language program will be the length of the program itself plus three sets of two-byte pointers giving the beginning address, the length of the program, and the starting or running address of the program. Some programs will add six bytes for these three pointers when calculating length.

LENGTH gives only the size of the running program without these pointers. If you compare the size given by this program with other programs, you may come up six bytes short depend-

Listing 1. Z80 Assembly Code for LENGTH.CO

```
00000 ;*********************************
00110 ;*                               *
00120 ;*         FILE LENGTH           *
00130 ;*         COMPUTATION           *
00140 ;*                               *
00150 ;*              BY               *
00160 ;*       THOMAS L. QUINDRY       *
00170 ;*                               *
00180 ;*       OCTOBER 26, 1985        *
00190 ;*                               *
00200 ;*********************************
00210          ORG     63145           ;0F6A9H FILE END
                                        = 0F76EH
00260 START    LD      HL,(POINT)
00270          CALL    20D5H           ;INCR BY 11 AND
                                        CHECK
00280          LD      (POINT),HL
00290          JP      NZ,NEXT         ;FOR VALID FILE
00300          LD      HL,6BF4H        ;POINTS TO
                                        STRING "BASIC"
00310          PUSH    HL
00320          LD      HL,(63898)      ;0F99AH BASIC
                                        BUFFER PTR
00330          EX      DE,HL
00340          CALL    BASIC4
00350          LD      HL,BUFFER
00360          PUSH    HL
00370          LD      HL,(63909)      ;0F9A5H PASTE
                                        BUFFER PTR
00380          PUSH    HL
00390          POP     DE
00400          CALL    DOC             ;TREAT LIKE A
                                        .DO FILE
00410          LD      HL,HIMEM
00420          PUSH    HL
00430          LD      HL,(62964)      ;0F5F4H HIMEM
                                        POINTER
00440          CALL    NUM3            ;HL IS VALUE
00450          LD      HL,FILES        ;MAXFILES
00460          PUSH    HL
00470          LD      HL,(64642)      ;0FC82H MAXFILES
                                        POINTER
00480          XOR     A
00490          LD      H,A
00500          CALL    NUM3
00510          LD      HL,1808H        ;H=COL 24,
                                        L=ROW 8
00520          LD      (0F639H),HL     ;POSITION LCD
                                        CURSOR
00530          CALL    7EACH           ;GET BYTES FREE
00540          CALL    12CBH           ;WAIT FOR
                                        KEYBOARD CHAR
00550          JP      0000            ;ANY KEY RETURNS
                                        TO MENU
```

```
00570 NEXT    LD      A,(HL)
00580         PUSH    AF
00590         CALL    344AH           ;PROGRAM ADDRESS
                                       IN DE
00600         POP     AF              ;GET FILE TYPE
00610         PUSH    HL              ;SAVE FILENAME
                                       POINTER
00620         LD      BC,START        ;SET UP STACK
                                       FOR RET
00630         PUSH    BC
00640         PUSH    DE
00650         POP     HL
00660         CP      192             ;IS IT A .DO?
00670         JP      NZ,ML
00680 DOC     LD      A,(HL)
00690         INC     HL
00700         CP      1AH             ;.DO FILES END
                                       IN 1AH
00710         JP      NZ,DOC
00720         JP      NUM2
00730 ML      CP      160             ;IS IT A .CO?
00740         JP      NZ,BASIC2
00750         INC     HL
00760         CALL    5AE3H           ;LD HL,(HL)
00770         JP      NUM3            ;HL HAS VALUE
00780 BASIC2  CP      128             ;IS IT A .BA?
00790         RET     NZ
00800 BASIC4  PUSH    DE              ;SAVE FILE START
00810 BASIC1  EX      DE,HL           ;MUST BE A .BA
00820 BASIC3  LD      E,(HL)
00830         INC     HL
00840         LD      D,(HL)
00850         LD      A,D
00860         OR      E
00870         JP      NZ,BASIC1       ;CHECK FOR DE=00
00880         POP     DE              ;DE HAS PROGRAM
                                       BEGIN
00890         DEC     DE              ;HL HAS PROGRAM
                                       END
00900 NUM2    LD      A,L             ;SUBTRACT DE
                                       FROM HL
00910         SBC     A,E
00920 NUM1    LD      L,A
00930         LD      A,H
00940         SBC     A,D
00950         LD      H,A
00960 NUM3    POP     BC
00970         POP     DE
00980         PUSH    BC
00990         LD      C,6             ;DISPLAY ENTRY
01000 PRINT   LD      A,(DE)          ;GET FILENAME
                                       CHARACTERS
01010         INC     DE
01020         RST     20H             ;DISPLAY
                                       CHARACTER
```

ing on how the other programs calculate the length.

HIMEM POINTER

When most machine language programs are run, there has to be memory set aside so that other programs can't corrupt it. The HIMEM pointer must be set to save this memory at the lowest running location of the program. For example, say a program runs from 62000 (decimal) to 62960, the highest normal user area of memory available. From BASIC, the command, CLEAR 50,62000 will reserve all user memory from 62000 up to but not including the value of MAXRAM, normally 62960.

If you forget that you have reserved this space and don't need it anymore, memory is wasted, as nothing but machine language programs can then use the space. My LENGTH program alerts you where the HIMEM pointer is set so that you can release this memory by the command, CLEAR 50,MAXRAM, if you don't need to keep it reserved. In the case of the HIMEM pointer, this program gives the location where HIMEM starts protecting memory. If no memory is protected, the HIMEM value will be 62960 unless another program has changed the value of MAXRAM.

MAXFILES

For each file buffer you have reserved using the MAXFILES=n command, 267 bytes are lost. This is probably the most obscure memory hog of all because most times the MAXFILES value is changed from within a BASIC program. To clear MAXFILES for the maximum use of memory, use the command, MAXFILES=0. This program gives you the number of MAXFILE buffers reserved.

THE PROGRAM

LENGTH cannot be considered even close to a memory hog itself because it is so short. Since it is short enough, the running location for LENGTH was chosen way up in that part of RAM normally used for the keyboard buffer. Starting at F685H (63109 decimal) for the next 255-258 bytes or so is the keyboard buffer. Allowing 36 bytes for keyboard input or whatever else that might take place (for insurance), LENGTH can easily be loaded into the keyboard buffer starting at location F6A9H (63145 decimal).

Since this memory location is above even the value of MAXRAM, no memory has to be reserved and this saves us from having to set the HIMEM pointer as in other machine language programs. It also means that we don't lose another 198 bytes for the program to run since it already uses a portion of memory that is not needed by anything at this time.

LENGTH is given here in two forms. The assembly code listing is given in Listing 1. Since Z80 code is inherently more popular among those who also have other TRS-80 computers, it is listed thusly.

Listing 2 gives a BASIC program that can be keyed directly into your Model 100. This BASIC program will poke the values given in the data statements directly into memory and then save the program as a .CO file. After it is saved, the BASIC program is no longer needed unless you want to upload the code to another computer.

TECHNICAL ASPECTS

To aid other programmers who may want to utilize some of the ROM calls and other vectors used in this program, Table 1 summarizes those aspects of ROM and vectors used by the operating system.

For a copy of this program and also SCRIPY.CO from this issue, send $6 for a 3.5-inch disk for the Tandy Portable Disk Drive to Thomas L. Quindry at 6237 Windward Drive, Burke, VA 22015. These programs have previously been placed into the Public Domain. □

```
01030           DEC     C
01040           JP      NZ,PRINT
01050           LD      A,20H
01060           RST     20H
01070           CALL    39D4H           ;CONVT AND
                                         DISPLAY NUMBER
01080           LD      A,(63034)       ;0F63AH CURSOR
                                         COL POS
01090           CP      27
01100           JP      C,MID           ;JP IF CURSOR
                                         COL <27
01110           JP      4222H           ;PRINT <CR><LF>
01120 MID       CP      14
01130           JP      C,BEGIN         ;JP IF CURSOR
                                         COL <14
01140           LD      A,27            ;SET CURSOR COL
                                         TO 27
01150           JP      CURSOR
01160 BEGIN     LD      A,14            ;SET CURSOR COL
                                         TO 14
01170 CURSOR    LD      (63034),A       ;0F63AH
01180           RET
01190 BUFFER    DEFM    'Buffer'
01200 HIMEM     DEFM    'Himem '
01210 FILES     DEFM    'Maxfil'
01220 POINT     DEFW    0F9AFH          ;F9AFH+11 IS
                                         START
01230           END     START
```

Listing 2. LENGTH.BA - Program to create LENGTH.CO

```
10 CLS
20 PRINT:PRINT TAB(11) "Loading LENGTH.CO"
40 PRINT TAB(19) "By"
50 PRINT TAB(11) "Thomas L. Quindry":PRINT
60 FOR N=63145 TO 63342:READ A:POKE N,A:CK=CK+A:NEXT
70 READ A:IF CK<>A THEN PRINT"Checksum error, check
   DATA statements"
   :END
80 PRINT TAB(8) "LENGTH.CO has been saved"
90 PRINT TAB(8) "LENGTH.BA can be killed"
100 PRINT TAB(5) "Does not need Himem protection"
110 SAVEM "LENGTH.CO",63145,63342,63145
1000 DATA 42, 109, 247, 205, 213, 32, 34, 109, 247,
     194
1010 DATA 241, 246, 33, 244, 107, 229, 42, 154, 249,
     235
1020 DATA 205, 28, 247, 33, 91, 247, 229, 42, 165, 249
1030 DATA 229, 209, 205, 3, 247, 33, 97, 247, 229, 42
1040 DATA 244, 245, 205, 46, 247, 33, 103, 247, 229,
     42
1050 DATA 130, 252, 175, 103, 205, 46, 247, 33, 8, 24
1060 DATA 34, 57, 246, 205, 172, 126, 205, 203, 18,
     195
1070 DATA 0, 0, 126, 245, 205, 74, 52, 241, 229, 1
1080 DATA 169, 246, 197, 213, 225, 254, 192, 194, 13,
     247
1090 DATA 126, 35, 254, 26, 194, 3, 247, 195, 40, 247
1100 DATA 254, 160, 194, 25, 247, 35, 205, 227, 90,
     195
1110 DATA 46, 247, 254, 128, 192, 213, 235, 94, 35, 86
1120 DATA 122, 179, 194, 29, 247, 209, 27, 125, 155,
     111
1130 DATA 124, 154, 103, 193, 209, 197, 14, 6, 26, 19
1140 DATA 231, 13, 194, 51, 247, 62, 32, 231, 205, 212
1150 DATA 57, 58, 58, 246, 254, 27, 218, 75, 247, 195
1160 DATA 34, 66, 254, 14, 218, 85, 247, 62, 27, 195
1170 DATA 87, 247, 62, 14, 50, 58, 246, 201, 66, 117
1180 DATA 102, 102, 101, 114, 72, 105, 109, 101, 109,
     32
1190 DATA 77, 97, 120, 102, 105, 108, 175, 249, 27952
```

Wonder how you get from here to there?

The first integrated Macintosh to laptop file transfer solution.

If you have a laptop computer and a Tandy or Purple Computer portable drive, your Macintosh is just a click away! **MacDOS** gives your Macintosh the power to access the portable drive quickly and easily.

Manage your Data

With **MacDOS**, your Macintosh can read and even edit files created with your laptop. You can Open, Edit, and Print files on the portable drive as easily as regular Macintosh files. **MacDOS** even has a Copy All command that can copy all of your files from the portable drive to the Macintosh for easy backups.

F E A T U R E S

- Full editing capabilities for files less than 43K in size
- up to 4 file windows open at a time
- catalog window shows files on the portable drive

P R I C I N G

MacDOS and adapter $95

No Messy Cables

MacDOS comes complete with adapter cable and software. All you do is plug the portable drive into the Macintosh via the special **MacDOS** adapter cable*.

MacDOS is easy to use.

Like all great Macintosh software, **MacDOS** is easy to use. **MacDOS** gives you quick access to the data you want, when you want it.

Best of all . . .

MacDOS is available NOW.
Call toll free to have Traveling Software give your Macintosh **MacDOS**: the connection to the portable drive.

*To order **MacDOS** call toll free:*

1 800 343-8080

Or write: **Traveling Software**
North Creek Corporate Center
19310 North Creek Parkway
Bothell WA 98011
(206) 483-8088

MacDOS

Circle 15 on reader service card.

TRAVELING SOFTWARE®

Traveling Software is a trademark of Traveling Software, Inc.

*Macintosh Plus owners will need the Mini-8 to DB9 connector cable to use MacDOS.

Go, No Go

A big problem in data communication is timing, or *synchronization.* When binary information is being transferred between two computers, or from a computer to a peripheral like a modem or printer, both systems have to know exactly what to expect — and when.

Imagine the problem of tuning in on a serial data transmission. What you'd be intercepting would be and endless stream of binary ones and zeros; a series of wavering tones through a modem. How do you get your bearings? How do you know which one or zero starts an eight-bit byte? During routine serial communications, that's the purpose of the infamous *stop bits,* as indicated by the fourth character in the TELCOM setup string (M7I1E means one stop bit).

That takes care of binary synchronizations — but timing means more than that. Let's say that your Model 100 system is connected at 9,600 bits per second (bps) directly to a host computer, be it a mainframe or a local microcomputer. Let's also say your 100 is connected to a parallel printer — and you've pushed F5 to echo what comes in through the serial port to the modem. And let's further assume that your printer is an Epson FX-80, able to print 160 characters per second.

A 9,600 bit per second serial line is equivalent to over 1,000 characters per second. That means that the data's coming in six times faster than it's going out — or for every character printed, five are lost.

THE SEVEN-BIT SOLUTION

The makers of ASCII, the American National Standards Institute, developed a solution to this problem. Called the XON/XOFF protocol, this solution allows either party of the telecommunications session to cry "Hold on a minute" whenever necessary.

A brief refresher: ASCII is the American Standard Code for Information Interchange. It's a seven-bit code, with 128 binary possibilities. It's what determines that the capital letters A through Z are character numbers 65 through 90, for example.

Not all of the ASCII characters are alphanumeric; codes 1 through 31 are reserved for special commands, such as ring the terminal bell (code 7) or execute a page eject or screen clear (code 12). The carriage return, line feed and backspace codes (13, 10 and 8, respectively) are among the most common "control sequences" used.

It has become common to refer to control sequences as Ctrl-*x*, where *x* is a letter of the alphabet. For example, since the code to ring the bell is ASCII code 7, and G is the seventh letter of the alphabet, the bell code is Ctrl-G. Similarly, the carriage return is Ctrl-M, line feed is Ctrl-J and form feed is Ctrl-L.

The ASCII control sequence for "Hold on a minute" is Ctrl-S, or ASCII code 19. The code for "Okay, continue with transmission" is Ctrl-Q, ASCII code 21. Look familiar? Those are the key sequences that most online services, such as GEnie or CompuServe, tell you to use to pause while online. If you look in a reference manual with an ASCII table, you'll see that Ctrl-S is called *DC1* and Ctrl-Q is *DC3.* The *DC* stands for *Device Control;* it's a holdover from the early days of computer communications where DC1 through DC4 were used to send commands to other devices, usually printers, that were attached to terminals.

The XON/XOFF commands are purely a software function. The telecommuncations software on both computers has to know to recognize the Ctrl-S and Ctrl-Q, and act appropriately. On the Model 100, recognition of XON/XOFF can be turned on and off by use of the fifth character in the TELCOM status string. M7I1E *enables* XON/XOFF; M7I1D *disables* the function. The Model 100 and Tandy 200 default to XON/XOFF enabled; the Tandy 102 defaults to disabled mode.

But back to our problem of the 9,600 bps feed to a 160 cps printer. That's where the TELCOM software comes in. TELCOM maintains an internal *buffer,* or storage area for characters. For a few seconds, it's able to continue capturing the characters that come in from the RS-232 port. When the buffer becomes half full, TELCOM sends a frantic XOFF command to the remote computer, begging it to stop sending temporarily. The remote computer halts — although it might take a few seconds if it isn't constantly checking for the XOFF. Finally, the flow of incoming characters halts — hopefully before the buffer overflows — and the Model 100 continues printing from the buffer to the parallel printer. When the buffer's nearly empty, the Model 100 transmits the XON code, and the remote computer resumes where it left off.

Sound foolproof? It isn't. XON/XOFF, like most communication protocols, makes some unwritten assumptions. First, that both computers are using the protocol — and not all systems do, especially software that's custom-written. Second, both computers should be constantly monitoring the line for the XON or XOFF code.

Again, not all packages do — they may send the incoming traffic directly to a buffer without checking for the Ctrl-S or Ctrl-Q, or may only be expecting data at certain intervals. Third, both computers must react instantly to the XON or XOFF signal. Sometimes that doesn't happen either, if the sending computer is transmitting data in large chucks, or blocks. It may only process the XON or XOFF after each block is sent. If the receiving computer's buffer is large enough, that's fine — but there's a possibility of data overflowing the buffer and being lost — and nobody knowing until much later.

But that's where XMODEM and other error-checking protocols come in, taking over when XON/XOFF and parity checking leave off.□

— Alan L. Zeichick

SUPER ROM

Lucid Spreadsheet **Write ROM** **Database** **Outliner**

NOW YOU CAN REALLY HAVE IT ALL!

All on one ROM. Truly the finest four programs available for the Model 100 — guaranteed. Try it for 30 days. If you aren't blown away by the excellence return it for a full refund.

$199.95

The four best programs for the Model 100 all on one ROM. 32K of power without using any RAM for program storage. This is the PCSG Snap-In ROM that just presses easily into the little ROM socket in the compartment on the back. You access the four right from the main menu like built-ins.

Write ROM — the definitive word processor for the Model 100. Function key formatting or dot commands. Search and replace. Library feature — inserts words, phrases or whole documents into text from just a code. MAP lets you see a picture of your document. In all there are 60 features and functions. No one can claim faster operation. FORM lets you create interactive forms with on-screen prompts that you can answer from the keyboard. Nothing else for the Model 100 compares with the features of Write ROM. Exactly the same as the Write ROM sold as a single program. Infoworld says it "makes the Model 100 a viable writing unit . . . surpassed our highest expectations for quality and clarity."

Lucid Spreadsheet: This is the one PICO magazine says "blows Multiplan right out of the socket" and Infoworld performance rated as "excellent" and said "makes the Model 100 compute." Gives you features you cannot get with Lotus 123. Lets you build spreadsheets in your Model 100 that would consume 140-150K on a desktop. Program generating capability with no programming knowledge required. Variable column widths. Includes find and sort with function key control. It's fast, recalculates like lightning. No feature has been taken from the original, only new ones added.

Database: This is a relational data base like no other. You can do everything from mailing lists to invoices. No complicated pseudo-coding, you create input screens as simply as typing into TEXT. You are not limited by size; you can have as large an input screen as you wish. Prints out reports or forms, getting information from as many files as you like. Complete math between fields. Total interface with Lucid worksheets.

Outliner: Does everything that Thinktank does on a PC but a whole lot better. Includes a Sort for your headlines. Lets you have headlines of up to 240 characters. Has cloning, hoisting and sideways scroll up to 250 characters. Like Lucid, this one sets a new standard for outliners. This is the way to plan and organize your projects.

Present Lucid and Write ROM owners can upgrade for $100. If you have both it's $75.

As usual PCSG sells the Super ROM on a thirty day guarantee. If for any reason you are not satisfied, simply return it for a full refund.

We are excited about this product. Super ROM gives the Model 100 the true power of a desktop. No other multi-program ROM has software that compares. But don't take our word for it. We invite you to make that comparison yourself. Priced at $199.95 on Snap-In ROM.

Got stuck with somebody else's multi-ROM? We'll upgrade it for $100.

(214) 351-0564

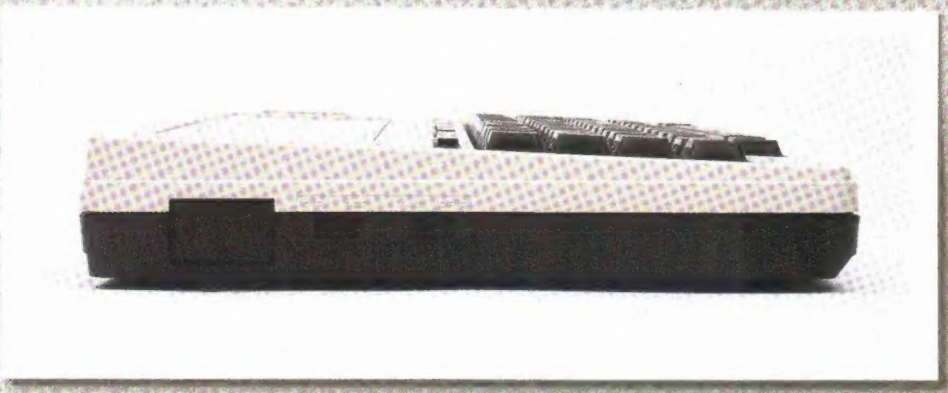
PORTABLE COMPUTER SUPPORT GROUP

11035 Harry Hines Blvd., #206, Dallas, TX 75229

MC, Visa, American Express, Check, or C.O.D.

Circle 40 on Reader Service Card.

Find the memory in these pictures.



Stock Tandy 102 with 24K of memory.

Tandy 102 with 1 megabyte of RAM Vault memory by P.G. Design

IF YOU NEED MORE MEMORY, YOU NEED THE VAULT!

The Vault adds as much as 1 megabyte of continuous RAM memory to the Tandy 102. With the Vault, you can maintain as many as 256 separate menu files (not just 19). It comes with four ROM sockets: 2 standard, 2 Radio Shack compatible. Adding only 3/8 inch of thickness, the Vault attaches firmly and easily with four screws.

For pricing information and technical questions, call us today!

313/727-2744

Circle 43 on Reader Service Card

DESIGN ELECTRONICS, INC.

37560 Thirty-one Mile Road, Richmond, MI 48062